Detection of Protein Modifications by Noise Model Based Analyses of Regulatory Information

Von der Carl-Friedrich-Gauß-Fakultät
Technische Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung des Grades
Doktor-Ingenieurin (Dr.-Ing.)

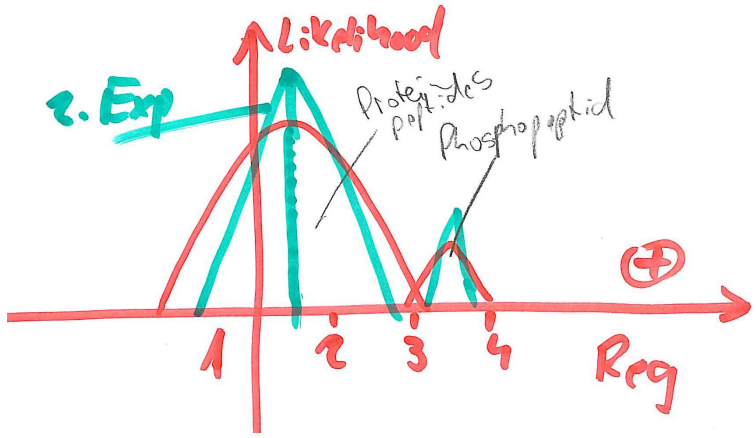
genehmigte

Dissertation

von Claudia Hundertmark
geboren am 26.04.1976
in Berlin

Eingereicht am: 11.11.2008
Mündliche Prüfung am: 18.12.2008
Referent: Prof. Dr. F. Klawonn
Korreferent: Prof. Dr. J. Wehland
Korreferent: Prof. Dr. H.-D. Ehrich

(2009)


Likelihood
2. Exp
Protein-des pept.-des
Phosphopeptid
+
1
2
3
4
Req

Contents

Zusammenfassung

In der quantitativen Proteomforschung werden durch massenspektrometrische Verfahren die vorhandenen Mengen einzelner Peptide und Proteine in unterschiedlich behandelten Zellen miteinander verglichen. Dabei kommt es zu Messungenauigkeiten, welche die Ergebnisse und somit die Hypothesenbildung verfälschen können. Davon betroffen sind hauptsächlich niedrige Signalintensitäten, bei welchen der Anteil des Rauschens einen signifikanten Anteil der gesamten Signalintensität ausmachen kann.

In der vorliegenden Arbeit ist es gelungen, das beobachtete Rauschen innerhalb eines difinierten Analyseablaufes mit Hilfe eines spezifischen Rauschmodelles zu beurteilen. Das Modell ermöglicht eine der Glaubwürdigkeit entsprechende Berechnung einzelner Peptidregulationsfaktoren sowie eine gewichtete Berechnung von Regulationsfaktoren für eine Gruppe von Peptiden, z.B. alle Peptide eines Proteins. Die so abgeleitete regulatorische Information wird durch Likelihoodkurven visualisiert, welche die Likelihood für den wahrscheinlichsten sowie alternative Regulationsfaktoren darstellen. Anhand der Gestalt einer Likelihoodkurve kann auf die Robustheit der zu Grunde liegenden Daten geschlossen werden.

Da die Entdeckung neuer post-translationaler Modifikationen essentiell für das Verständnis dynamischer Proteinnetzwerke ist, sind quantitative massenspektrometrische Analysen auf der Peptidebene derzeit Ziel vieler biologischer Projekte. Modifizierte Peptide sind häufig nur in sehr geringen Mengen vorhanden, daher ist die Beurteilung der Robustheit besonders für diese Peptide von großem Interesse. Wenn ein Peptid modifiziert wird, nimmt korrespondierend die Menge seiner unmodifizierten Form ab. So kann gelegentlich beobachtet werden, dass diese im Massenspektrum neben dem modifzierten und in eine Richtung regulierten Peptid vorhanden und dann oft in in die Gegenrichtung reguliert ist. Da mittels Massenspektrometrie nur nach einer oder sehr wenigen der über 200 beschriebenen Arten von Modifikationen gleichzeitig gesucht werden kann, ist die Detektion von differentiell regulierten Peptiden innerhalb eines Proteins von größtem Interesse, um so auf potentielle neue Modifikationen schließen zu können. Zu diesem Zweck ist in der vorliegenden Arbeit neben der Berechnung der regulatorischen Information ein Clusteringalgorithmus entwickelt worden, welcher (auf dieser basierend) nach differentiell regulierten Peptiden eines Proteins sucht.

Summary

In quantitative proteomics the amounts of individual peptides and proteins within differentially treated cells are compared by mass spectrometry. Occuring impreciseness of the measurements can adulterate the results and thus, formulation of hypotheses. Especially low signal intensities are affected since considerable percentages of those may be caused by noise.

In this work, the observed intensity dependent noise within a defined quantitative mass spectrometry based workflow could be modelled by the development of a specific noise model. Both calculation of regulation factors of single peptides and calculation of such of peptide groups (e.g. all peptides identified within one protein) is derived from the noise model. In doing so, all calculations are weighted according to the robustness of the underlying data. The regulatory information obtained in this way, is visualised by likelihood curves presenting the likelihood of the most probable as well as alternative regulation factors. The reliability of the most suitable regulation factor – and consequently the robustness of the data – can be inferred from the shape of the curves.

As the detection of novel post-translational modifications (PTM) is essential for the understanding of dynamic protein networks, many biological projects currently aim on quantitative analyses by mass spectrometry on the peptide level. Often, the abundances of modified peptides are very low and therefore, the statistical evaluation of the regulatory information is of highest importance regarding such peptides. During modification of a peptide, the amount of the unmodified peptide decreases correspondingly. Thus, in mass spectra not only the modified and optionally regulated peptide but also the unmodified variant of the same peptide – regulated contrary – can be observed. The detection of PTM by mass spectrometry is restricted to just a few out of more than 200 different kinds of modifications at the same time. Consequently, the identification of differentially regulated peptides within the same protein is highly interesting for the investigation of new peptide modifications. For this purpose, besides calculation of regulatory information a clustering algorithm was developed in this work that is able to find differentially regulated peptides of a protein.

1 Introduction

Increasing amounts of data generated by today's high-throughput technologies require enhanced strategies for the interpretation and handling of data. Besides optimised strategies for data storage e.g. by databases and data warehouses concepts from machine learning and data mining are introduced into analysis of high-throughput data, e.g. genomics, transcriptomics, metabolomics and proteomics.

The dominant 'omics' field during the last decade was genomics, which addresses the genome sequence including the genes, their structure and encoded functional information. Meanwhile, over 700 bacterial and 22 eukaryotic genomes including the human genome comprising 3.000.000.000 basepairs (bp) were completed [1]. Transcriptomics studies the set of all messenger RNA molecules ("transcripts") of one population of cells. Hundreds or thousands of genes are analysed regarding their expression often using high-throughput techniques based on DNA microarray technology. The metabolome represents the set of all metabolites – intermediates and products of metabolism – in an organism. Thus, metabolomics is the quantitative analysis of metabolites often using approaches from mass spectrometry which is one of the main techniques in proteomics as well. Proteomics aims at the identification and representative characterisation of all proteins in a cell under defined conditions (proteome). Like the transcriptome and the metabolome, the proteome is highly dynamic and varies significantly regarding its qualitative and quantitative composition during the cell cycle and changing environmental conditions.

Objectives of this Work Proteins and their fragments (peptides) are analysed quantitatively by application of a mass spectrometry approach (LC-MS/MS) joined with one of the available labelling techniques (e.g. iTRAQ^{TM}). Similar to most measurements, quantitative analysis of peptides and proteins using iTRAQ^{TM} is corrupted by noise. Usually, those imprecision does not influence a high signal significantly. Low intensities, however, can be highly affected by such additional intensities. As a possible consequence the observed regulation factor does not correlate with the real relative abundances of the investigated objects

[1] http://www.ncbi.nlm.nih.gov/genomes/static/gpstat.html, 17.10.08

or, as an extreme example, the information suggests even an opposite and wrong direction of regulation.

Regarding signal transduction studies and post-translational modifications the amounts of available peptides often are very small and therefore, small intensities are highly important for the detection of regulations and post-translational modifications. Thus, small intensities that are potentially strongly affected by noise, can not be discarded and consequently, the evaluation of their reliability is requested. An approach for the analysis of the robustness of those data is to apply a noise model reflecting the likelihood of the calculated regulation as well as the likelihood of alternative regulations. When this information can be identified and realised in an intuitive manner, the reliability of regulatory information of peptides and proteins can easily be evaluated. Such a strategy would certainly support the detection of post-translational modifications.

Organisation of this Work This work is organised as follows: Section 2 gives an overview of proteomics. Besides proteins and post-translational modifications of those it focuses on the technique for measuring proteins and peptides quantitatively. Subsequently, the observed noise of the measurement is described followed by the presentation of a specific noise model. After parameter estimation the model assumptions and the parameter estimation are validated (section 3). Section 4 presents both a new approach for the calculation and visualisation of regulatory information based on the established noise model as well as a resulting software tool. A clustering strategy for the detection of unknown post-translational modifications by the identification of differently regulated peptides within one protein is introduced in section 5. This strategy is applied to an experimental dataset exhibiting the correctness and potential of this approach. Finally, section 6 summarises the results and provides an outlook on the future work.

2 Proteomics

The proteome – any proteins in a cell under defined conditions – is highly dynamic and varies significantly regarding its qualitative and quantitative composition during the cell cycle and changing environmental conditions. Besides identification proteomics allows relative quantification of proteins and their fragments (peptides) in cells using approaches from mass spectrometry. In the following, an introduction to proteomics and the applied techniques is given which is necessary to understand the biological background of this work.

2.1 Proteins

Proteins are the main actors within the cell and are found in all living systems ranging from bacteria to higher mammals such as humans. Proteins are present in greater amounts than any other biomolecule – more than 50% of the dry weight of cells consists of them. Many of them are enzymes catalysing biochemical reactions, others are regulatory proteins that contribute to the correct expression of the genome. Some proteins are carriers moving molecules within or between cells and others are structural proteins building cellular components. While the genome is essentially identical in all cells within an organism, the set of expressed proteins varies extensively through time in order to realise cellular function under variable environmental conditions.

2.1.1 Composition of Proteins

Proteins are large organic compounds composed of 20 different amino acids which are arranged in a chain. All types of amino acids have common structural features, including a carbon to which an amino group, a carboxyl group, a hydrogen and an amino acid specific side chain are bonded (Figure 2.1[1]). The different side chains account for the differing properties of amino acids, e.g. hydrophobicity and charge. Consequently, the sequence of amino acids determines both biochemical properties and function of a protein.

[1] taken from http://en.wikipedia.org/wiki/, 16.09.08

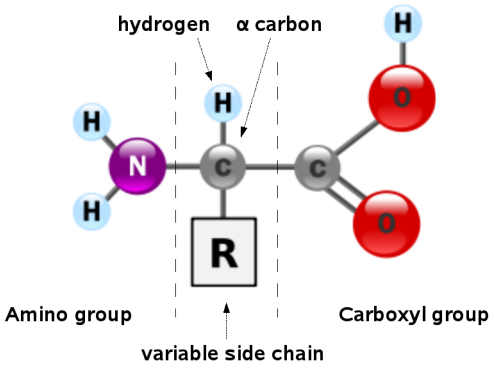


Figure 2.1: General structure of amino acids. R represents a side chain which is specific to each amino acid.

Amino acids can be abbreviated either by 3-Letter-Code (3LC) or by 1-Letter-Code (1LC), which are listed in Table 2.1. The last column gives the mass of the amino acids normally reported in units of daltons. One dalton is defined as $\frac{1}{12}$ the mass of carbon (^{12}C). Protein sequence lengths fluctuate between about hundred and many thousands of amino acids.

Table 2.1: Abbreviation of amino acids: 3-Letter-Code (3LC), 1-Letter-Code (1LC) as well as specific masses.

Amino Acid	3LC	1LC	Mass [Da]	Amino Acid	3LC	1LC	Mass [Da]
Alanine	Ala	A	71.0	Leucine	Leu	L	113.1
Arginine	Arg	R	156.1	Lysine	Lys	K	128.1
Asparagine	Asn	N	114.0	Methionine	Met	M	131.0
Aspartic acid	Asp	D	115.0	Phenylalanine	Phe	F	147.1
Cysteine	Cys	C	103.0	Proline	Pro	P	97.1
Glutamic acid	Glu	E	129.0	Serine	Ser	S	87.0
Glutamine	Gln	Q	128.1	Threonine	Thr	T	101.0
Glycine	Gly	G	57.0	Tryptophan	Trp	W	186.1
Histidine	His	H	137.1	Tyrosine	Tyr	Y	163.1
Isoleucine	Ile	I	113.1	Valine	Val	V	99.1

The individual amino acids are linked by peptide bonds. As both the amine and carboxylic acid groups of amino acids can react to form amide bonds, one amino acid molecule can react with another one and join through an amide linkage. In this process one molecule of water (H_2O) is released. The resulting CO–NH bond is called a peptide bond. Formation of a peptide bond is shown in Figure 2.2.

Figure 2.2: Formation of a peptide bond: one amino acid molecule can react with another one and join through an amide linkage. In this process one molecule of water (H_2O) is released. The resulting CO–NH bond is called a peptide bond.

Short sequences comprising less then 50 amino acids are usually termed peptides. Figure 2.3[2] illustrates a short peptide composed of five amino acids which are linked by peptide bonds (green). Each of the amino acids has a side chain R_x, the end of the sequence with a free amino group (left hand side, red) is termed the N-terminus (amino terminus), whereas the end with a free carboxyl group (right hand side) is known as the C-terminus (carboxy terminus, red).

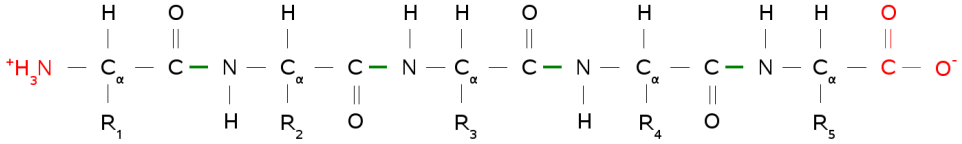


Figure 2.3: Peptide consisting of five amino acids which are linked by peptide bonds (green). Side chains are presented by R_x, N-terminus (left hand side) and C-terminus (right hand side) are denoted in red.

Protein sequence lengths fluctuate between several hundreds and many thousands of amino acids. The total mass of a protein is added up by the mass of the comprising amino acids and is reported in dalton (Da) or kilodalton (kDa).

2.1.2 Protein Function

The function of a protein is based on its 3D structure which is determined by the amino acid sequence and modification on the amino acids. All cellular functions require proteins which for example are responsible for transportation processes, work as hormones or act as antibodies for infection defence. Besides structural proteins, the majority of proteins are enzymes and mediate nearly every type of cellular function by controlling metabolic and signalling networks.

2.1.3 Modifications of Proteins

Different modifications occur and extend the range of protein functions. Thus, from the same genetic locus different proteins are generated (alternative splicing).

[2] taken from Lodish *et al.* (1996)

Also post-translational modifications (PTM) are able to alter the protein function significantly. Often, PTM regulate molecular interactions of different proteins.

Post-translational modifications are modifications of proteins after the initial synthesis of the whole protein (translation). Currently, about 200 different types of modifications (Walsh *et al.*, 2005) were described at proteins altering their structure, activity state, localisation, or stability. Hence, the range of functions of the protein is extended by modifications accounting for removing the terminal part of the sequence, adding new groups, and modifying existing groups such as the oxidation of thiol groups. When the thiol groups of two cysteine residues are taken close to each other in the course of protein folding, an oxidation reaction can create a cystine unit with a disulfide bond (-S-S-) which can contribute to structural changes of a protein. Other modifications, like phosphorylation, are part of common mechanisms for controlling the behaviour of a protein, for instance activating or in-activating an enzyme.

2.1.4 Phosphorylation of Proteins

Phosphorylation of eucaryotic proteins can be detected as additional phosphate group (PO_4^{3-}) to the amino acid residues serine (S), threonine (T), and tyrosine (Y). Reversible phosphorylation of proteins is an important regulatory mechanism which enzymes named kinases and phosphatase account for. Enzymes themselves are often activated and deactivated by reversible phosphorylation. Adding and removing a phosphate molecule strongly affects a protein's polarity and structure. In this way conformational changes in the structure of the protein are possible as well as protein interactions by specific domains.

During phosphorylation phosphate groups from the high-energy molecule ATP are transferred to specific target proteins by kinases. On the contrary, in the case of dephosphorylation phosphate groups are removed from their substrates by phosphatases. Using any peptides as example phosphorylation and dephosphorylation reaction is exemplified in Figure 2.4.

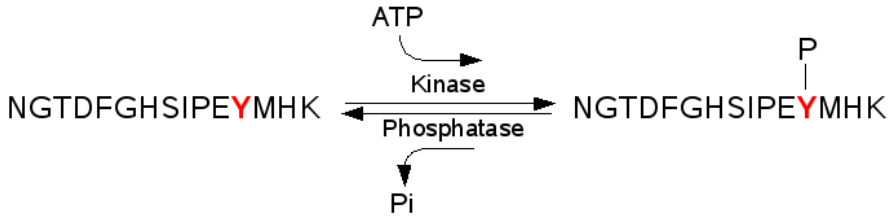


Figure 2.4: Phosphorylation and dephosphorylation reaction. The peptide `NGTDFGHSIPEYMHK` is phosphorylated by a kinase and dephosphorylated by a phosphatase at the amino acid tyrosine (Y).

2.2 Mass Spectrometry

Mass spectrometry (MS) is used for measuring the masses of the molecules. However, mass spectrometers are only able to recognise charged molecules, therefore the molecules must be ionised. In proteomics the ionisation is commonly achieved by the addition of protons, and more rarely by loss of protons. Hence the mass of the peptide or protein is increased and decreased by the nominal mass of 1 Da multiplied by the number of charges (protons) in the case of addition and loss of protons, respectively. By convention the number of added and lost protons is denoted by z resulting in positively and negatively charged ions. From the measured mass/charge ratio (m/z) of ions the molecular masses can be determined allowing to identify the molecular composition of a given sample of analyte. In proteomics, the analyte is usually a collection of peptides derived from a protein sample by digestion with a protease like trypsin.

2.2.1 Instrumentation

Mass spectrometers consist of three basic components: an ion source, a mass analyser and an ion detector. Ions are produced by transformation of the molecules in the sample into ionised fragments. The ions are accelerated in an electric field towards the analyser which separates the ions according to their mass/charge ratio. The function of the detector is to record presence and "number" of individual ions.

Depending on the kind of ion source and mass analyser several different types of mass spectrometers can be distinguished. The instrument which was used for the generation of all data presented in this work, is explained in detail in the following.

Electrospray Ionisation Quadrupole Time Of Flight Mass Spectrometer In the case of an Electrospray Ionisation Quadrupole Time Of Flight mass spectrometer (ESI-QTOF) the ionisation is achieved by electrospray ionisation of appropriate solvents containing the analytes. The analyte is dissolved and forced through a narrow needle held at a high voltage. A fine spray of charged droplets emerges from the needle and is directed into the vacuum chamber of the instrument. Entering the mass spectrometer, the droplets are dried using a stream of gas, resulting in gas-phase ions.

The mass analyser consists of several quadrupole sections and an additional time of flight analyser. A quadrupole is a set of four parallel metal rods with an electric field in between. It can be operated in different modes, either allowing ions of any m/z ratio to pass through, or in scanning mode, where a potential difference is applied and the instrument acts as a mass filter. In the latter case, ions of a selected m/z ratio are allowed to pass through to the detector whereas all

others are deflected from their linear flight path and eliminated from subsequent analysis. By varying the voltage over time, ions with different m/z ratios can be selectively allowed through to the detector and a mass spectrum of the analyte can be obtained. The included TOF analyser exploits the fact, that heavy ions will take longer to travel down a field-free tube than lighter ones if all ions are carrying the same charge.

Finally, the detector records the presence and "number" of individual ions. With the aid of software the measurements are interpreted resulting in a mass spectrum (see section 2.2.2, Figures 2.6 and 2.8).

2.2.2 Proteomics Approaches

Generally, proteomics approaches are divided into two types of strategies: top-down proteomics and bottom-up proteomics. In the top-down approach, intact proteins are directly used for the analysis and identified by its mass whereas in the widespread bottom-up approach, proteins are first cleaved into peptides. These peptides are introduced into the mass spectrometer and identified by peptide mass fingerprinting or tandem mass spectrometry. Hence, bottom-up proteomics uses identification at the peptide level to infer the existence of proteins.

Peptide Mass Fingerprinting Peptide mass fingerprinting (PMF) or peptide mapping refers to the identification of proteins using measured peptide masses. The principle of this technique is that each protein can be identified by the masses of its peptides. For the generation of peptides from the proteins of the sample, the proteins have to be digested with a specific cleavage reagent, usually trypsin. Trypsin predominantly cleaves peptide chains at the carboxyl side of the amino acids lysine (K) and arginine (R) as illustrated in Figure 2.5.

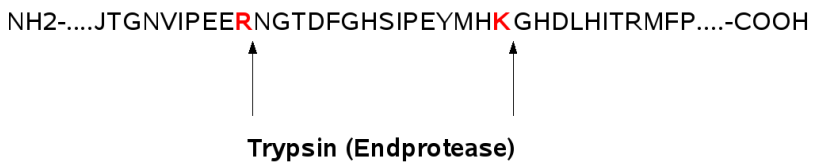


Figure 2.5: Cleavage sites of a polypeptide chain by trypsin which cleaves peptide chains at the carboxyl side of the amino acids lysine (K) and arginine (R).

Subsequent to the measurement the obtained peptide masses are compared with theoretical peptide masses of known proteins (in-silico digest) which are derived from the amino acid masses (see Table 2.1). Indeed, the obtained peptide masses do not yield in unique identification of the peptides since peptides containing equal

amino acids in different orders have the same masses. Even completely different peptides sometimes have very similar masses that are not able to be distinguished perfectly. Figure 2.6 shows a spectrum obtained by peptide mass fingerprinting. Different peptides are identified by different detected masses. The x-axis refers to m/z, the y-axis refers to the intensity which was measured by the detector. Each peak represents a particular peptide from the protein. It should be mentioned, that due to different peptide properties the peak heights do not correlate with the amounts of different peptides in PMF.

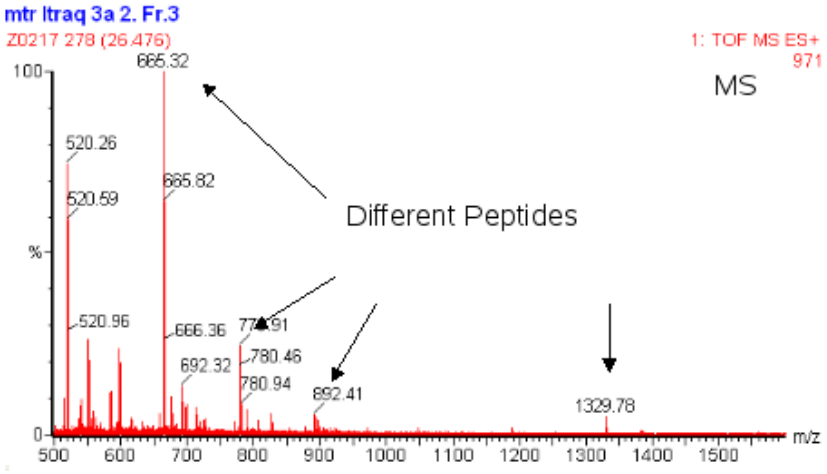


Figure 2.6: Mass spectrum containing several different peptides obtained by peptide mass fingerprinting. The x-axis refers to m/z, the y-axis refers to the intensity which was measured by the detector. Each peak represents a particular peptide from the protein.

Peptide Sequencing by Tandem Mass Spectrometry Spectra of peptides generated by tandem mass spectrometry (MS/MS) may provide significant information for the determination of the amino acid composition within the peptide (peptide sequencing) in contrast to spectra generated by MS. By using two mass analysers in series separated peptides fragment at predetermined breaking points into corresponding ions (Figure 2.7). Depending on the used MS device ions from particular series are more or less frequent.

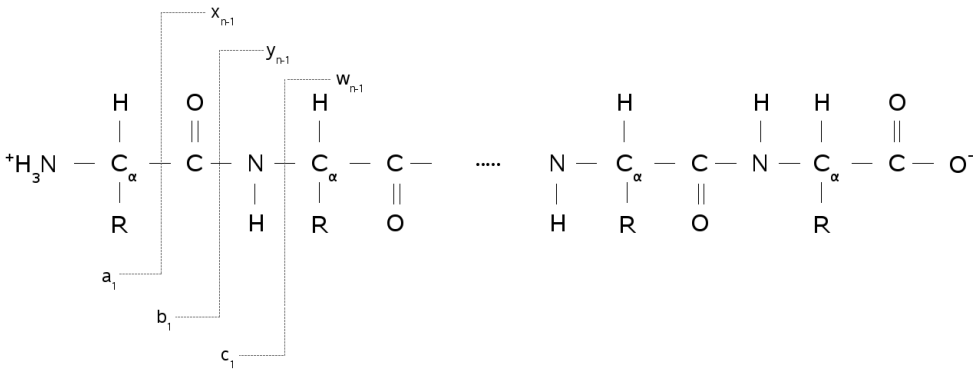


Figure 2.7: Breaking points of peptides during fragmentation in MS/MS. Different kinds of ions can be generated. Depending on the breaking point the ions are called a/x, b/y and c/w ions.

The amino acids of a peptide can be derived from the measured fragment ion masses. The correct order of the total or at least parts of the sequence are determined by comparing detected masses and known amino acid masses, considering that ions may contain more than one amino acid (Figure 2.8). Nowadays, this work combined with comparisons of the obtained results with theoretically peptides is done routinely by software.

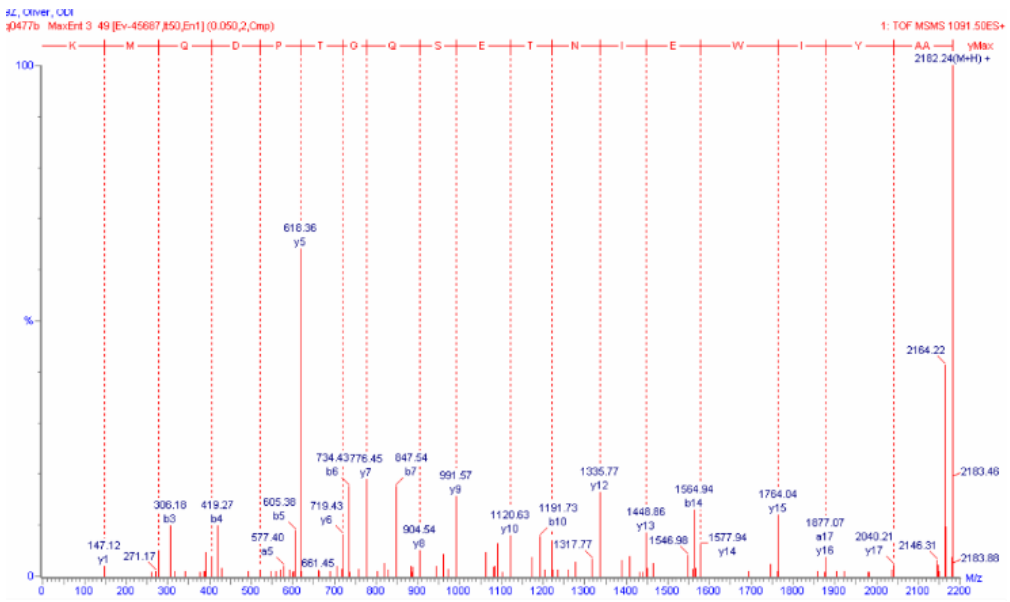


Figure 2.8: Peptide sequencing: Annotated peptide mass spectrum generated by MS/MS. Each amino acid of the peptide is represented by specific mass (m/z, x-axis).

2.2.3 Combination of Mass Spectrometry and Chromatographic Techniques

In the case of complex samples containing more than one or a few proteins, separation of the analyte is helpful. Different chromatographic separation techniques are recommended depending on the analysed compounds. Before the peptides are subsequently introduced into the mass spectrometer, liquid chromatography (LC) is performed in order to separate the peptides (LC-MS/MS).

2.3 Quantification of Peptides and Proteins

Besides identification, mass spectrometry can be used for relative and absolute quantification of proteins. Comparative analyses are performed in order to reveal proteins regulated under specific conditions and to define involved networks, processes and signalling pathways. LC-MS/MS typically investigates protein-derived peptides. Thus strategies for the quantification and characterisation of proteins preferentially should assess the peptide level.

iTRAQ™ (Isobaric Tag for Relative and Absolute Quantitation) – introduced by Ross *et al.* (2004) – became one of the standard techniques for relative quantifications of automatically sequenced peptides. iTRAQ™ allows relative as well as absolute peptide quantification of up to eight different samples in parallel using up to eight differential labelling reagents (Pierce *et al.*, 2008). During the labelling process the selected type of iTRAQ™ molecules is covalently linked to every peptide from one biological sample. All eight iTRAQ™ molecules have the same structure and molecular weight, but differ in the distribution of incorporated isotopes (Figure 2.9). In the intact molecule the total mass is balanced by the so-called balancer group and each labelling reaction introduces an identical mass shift. However, under the conditions of peptide sequencing (MS/MS) iTRAQ™ also produces fragment ions that differ in mass and serve as sample specific reporters: Same peptides (with identical amino acid sequence) from different biological samples which were labelled differentially and are subsequently pooled exhibit the same biochemical properties and total masses. Consequently, identical peptides from different samples co-elute at the same time from chromatographic columns, enter with the same molecular weight the MS device and are subjected commonly to the fragmentation process. The ratios of the released iTRAQ™ reporter ions correlate with the relative abundance of the analysed peptides as part of the investigated samples.

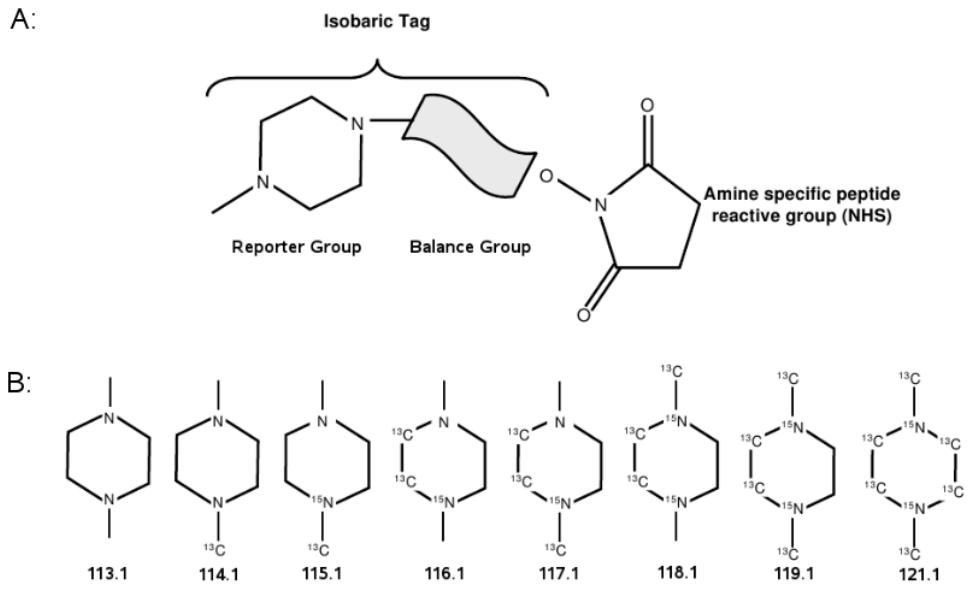


Figure 2.9: Chemical constitution of the iTRAQ^{TM} molecules (taken from Pierce *et al.* (2007)). A: Reporter group, balance group and reactive group. B: Eight variants of iTRAQ^{TM} molecules resulting in total masses 113.1 – 119.1 Da and 121.1 Da.

A typical iTRAQ^{TM} workflow allowing the relative quantification of peptides derived from proteins of two different samples is summarised in Fig. 2.10. Initially, all proteins from both samples are cleaved into peptides by a specific endo-protease (e.g. digestion with trypsin). Thereafter, both peptide fractions are labelled separately using different iTRAQ^{TM} reagents each containing reporter groups of different masses (e.g. 114.1 Da or 116.1 Da). iTRAQ^{TM} molecules are linked covalently to the N-terminus of each peptide as well as to every present lysine in the peptide sequence. Identical peptides from different samples exhibit a modified but identical chemical behaviour and mass subsequent to the iTRAQ^{TM} labelling. Therefore, differentially labelled samples can be pooled before the MS analyses. Whereas the amino acids of peptides from both samples commonly contribute to the total ion intensities used for the peptide sequencing, the reporters dissociate in sample-specific amounts. Reporter ions with specific masses (113.1 Da, 114.1 Da, 115.1 Da, 116.1 Da, 117.1 Da, 118.1 Da, 119.1 Da, 121.1 Da, see Figure 2.9) can be detected as part of every peptide fragmentation spectrum and a direct comparison of their intensities gives information about the relative abundance of peptides in the compared samples.

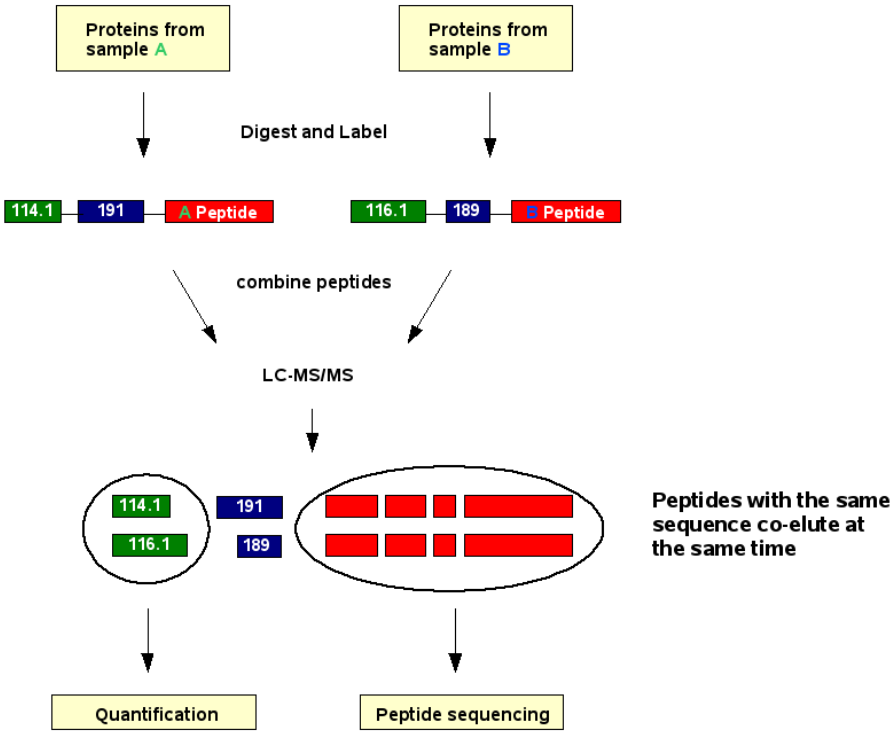


Figure 2.10: iTRAQ^{TM} workflow: proteins from samples A and B are digested, peptides (red) are iTRAQ^{TM} labelled (tags 114.1 and 116.1) and combined. When performing LC-MS/MS peptide bonds between the amino acids as well as bonds between iTRAQ^{TM} molecules and peptides are broken. Subsequently, peptides are used for identification and iTRAQ^{TM} molecules are used for relative quantification.

Figure 2.11 shows reporter intensity peaks of one peptide from two differentially labelled samples within an MS/MS spectrum. The iTRAQ^{TM}-reporters 114.1 and 116.1 are used for peptide labelling and are detected with most intense peaks. Small signals at m/z 115.1 and 117.1 are caused by specified impurities of the reagent. Regarding the percentage intensities on x-axis the peptide of this spectrum was found about 20 times more frequently in the 116.1 labelled sample than in the 114.1 labelled sample.

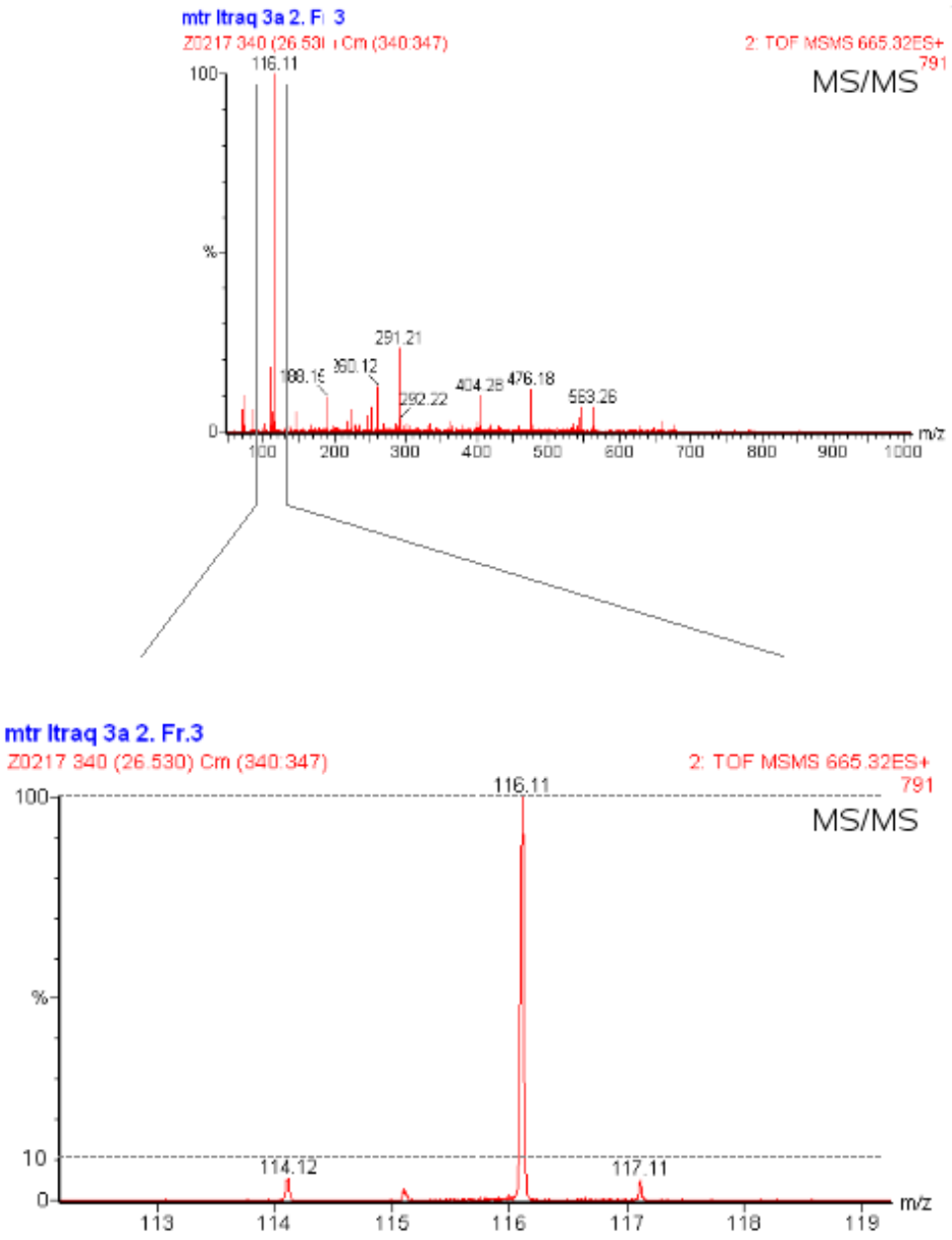


Figure 2.11: MS/MS spectrum of a peptide labelled with iTRAQ^{TM} reagents 114.1 and 116.1. Regarding the percentage intensities on x-axis the peptide is detected nearly 20 times more frequent in the 116.1 labelled sample than in the 114.1 labelled sample (extended reporter mass region in the lower part of the figure).

The upper MS/MS spectrum shows intensity peaks based on areas. Stick spectra are generated from those profile spectra by centering. Previous tests yielded that relative proportions are not changed by centering (data not shown). Regulatory information can be presented as ratios (regulation factor, expression ratio), which are calculated from pairwise comparisons of iTRAQ^{TM} reporter ions of peptides from all samples. According to the measured intensities of the peptide shown in

Figure 2.11 the regulation factor is about 20 ($\frac{1}{20}$) since the peptide is detected nearly 20 times more frequent in the 116.1 labelled sample than in the 114.1 labelled sample.

2.4 Experimental Data and Data Processing

Most of data presented in section 5 were generated from HGF stimulated human epithelial cells and untreated control cells (HGF/Met-activated signalling studies). Peptides were separated using a nanoAcquity HPLC (Waters Corp., USA) that was linked to a Q-TOFmicro mass spectrometer (Waters Corp., USA). Postprocessing of fragmentation data contained combination of 4 MS scans to one spectrum, smoothing based on Savitzki-Golay and generation of centroid spectra. MS data that were acquired and processed by MassLynx (version 4.1, Waters Corp., USA) were searched for protein identification using the UniProtKB/Swiss-Prot database (release 55.0 of 26-Feb 2008 with 356194 entries; taxonomy: Homo sapiens with 18610 entries) and Mascot Daemon 2.1.6. Only peptides that were unambiguously sequenced and identified (Mascot V 2.1, Matrix Science, Perkins *et al.* (1999)) were included in the statistical evaluation of regulatory data.

3 iTRAQ™-specific Noise Model

Often, measuring biological components generates results corrupted by noise. Noise can be caused by various factors as the detector itself, amplifier circuits, sample properties or data processing and appears independently of the applied technology (i.e. microarray, mass spectrometry). In contrast to microarray data examined in detail concerning their impairment by noise (e.g. Baldi and Long (2001)) such investigations of mass spectrometry data – especially quantitative data generated with the new technology iTRAQ™ – are just in the beginning. Noise models for microarray data are enjoying great popularity for some years already (e.g. Rocke and Durbin (2001); Tu *et al.* (2002); Weng *et al.* (2006)), whereas the development of more precise noise models for MS data has only started recently (Anderle *et al.*, 2004; Du *et al.*, 2008). Only a few studies regarding iTRAQ™ specific noise can be found in literature (Hu *et al.*, 2006; Lin *et al.*, 2006; Boehm *et al.*, 2007). In all cases, it was observed and assumed that noise depends on the measured intensity.

3.1 Data Preprocessing

Preprocessing is a necessary step before the data are used for model building or analysis. In the case of iTRAQ™ data preproccesing at least contains the correction of isotopic impurities and sample normalisation. Sample normalisation is investigated in-depth for several types of data (e.g. Bolstad *et al.* (2003)). Besides a few approaches concerning the development of iTRAQ™ specific noise models that introduce preprocessing, D'Ascenzo *et al.* (2008) presented a software package focusing on preprocessing and visualisation of 8-plex iTRAQ™ labelled data. Several transformations like the correction of isotopic impurities and different normalisation strategies can be performed user-defined.

Before analysing data in this work, an improved peak detection as well as isotopic impurity correction, sample normalisation and logarithmic transformation are applied to quantitative data postprocessed as described in section 2.4. From such data the actual iTRAQ™ reporters are read out using Mascot Parser 2.1.00 (Matrix Science), which is an object-oriented Application Programmer Interface (API) to both Mascot result and configuration files. In detail, the performed preprocessing items are:

1. Peak detection

   If only one signal corresponds to each iTRAQ™ reporter within a range of reporter mass ± 0.05Da, this signal is selected as reporter intensity. However, in case of multiple signals near the specified mass an optimised algorithm for peak detection is applied: The mass ranges of all used reporters are scanned for the existence of a mass pattern that contains signals comprising exact differences (1 Da) of the expected masses. Figure 3.1 illustrates this approach: Two signals are detected near the expected mass of the iTRAQ™ label 115.1. Even though the higher signal (mass ∼ 115.1 Da) matches better the expected mass, the lower signal (mass ∼ 115.05 Da) is selected by the algorithm. The signal ∼ 115.05 Da is more reliable since signal intensities comprising 1 Da mass differences are detected (114.05 Da, 116.05 Da, 117.05 Da). Obviously, a mass shift appeared and is compensated by this new algorithm for peak detection.

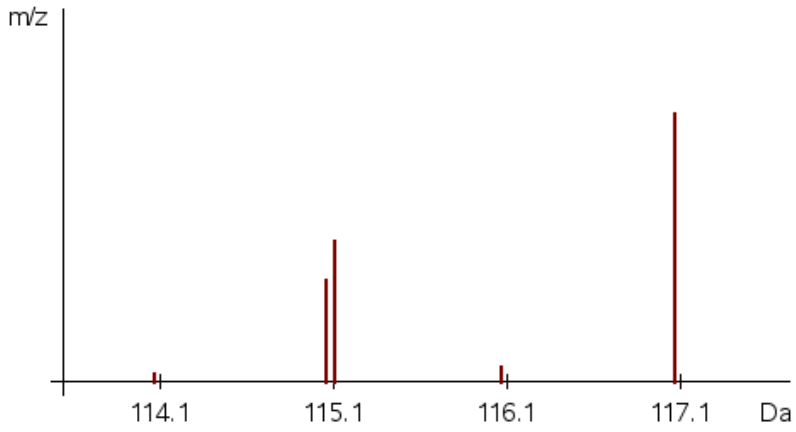


Figure 3.1: Multiple possible signals corresponding to iTRAQ™ label 115.1. Due to the presence of a pattern of 1 Da the lower intensity is selected by the optimised algorithm for peak detection.

2. Correction of isotopic impurities

   Each iTRAQ™ reporter type can contribute to the neighbouring signals with a few percent of its own ion intensity (isotopic impurity). A small fraction of each iTRAQ™ molecule batch is routinely subjected to individual fragmentation experiments in order to certify its signal specificities. Afterwards, the percentage of certified isotopic impurities of each applied iTRAQ™ molecule type is introduced in a linear system of equations. This allows to calculate precisely the actual ion abundances of differently labelled peptides.

Based on the measurement of signal intensities as given in Table 3.1 and transformations as exemplified with iTRAQ$^{\text{TM}}$ 114.1 in (3.1) a system of linear equations is obtained. Variables $a \dots d$ and indices $-2 \cdots +2$ refer to the percentages of the true total intensity ($i^{(c)}_{11x.1}$) and $i_{11x.1}$ corresponds to the measured intensity at mass $11x.1$ Da.

Table 3.1: Presentation of signal intensity measurements: Intensities of each iTRAQ$^{\text{TM}}$ reporter and its neighbouring masses in a distance of -2 Da, -1 Da, $+1$ Da and $+2$ Da are necessary for the calculation of isotopic impurities.

iTRAQ$^{\text{TM}}$ reporter	mass-2 Da	mass-1 Da	mass	mass$+1$ Da	mass$+2$ Da
iTRAQ$^{\text{TM}}$ 114.1	a_{-2}	a_{-1}	a_0	a_{+1}	a_{+2}
iTRAQ$^{\text{TM}}$ 115.1	b_{-2}	b_{-1}	b_0	b_{+1}	b_{+2}
iTRAQ$^{\text{TM}}$ 116.1	c_{-2}	c_{-1}	c_0	c_{+1}	c_{+2}
iTRAQ$^{\text{TM}}$ 117.1	d_{-2}	d_{-1}	d_0	d_{+1}	d_{+2}

Using iTRAQ$^{\text{TM}}$ 114.1 as an example, the performed transformations are

$$
\begin{aligned}
& i^{(c)}_{114.1} \\
&= i_{114.1} + a_{-2}i^{(c)}_{114.1} + a_{-1}i^{(c)}_{114.1} + a_{+1}i^{(c)}_{114.1} + a_{+2}i^{(c)}_{114.1} - b_{-1}i^{(c)}_{115.1} - c_{-2}i^{(c)}_{116.1} \\
&= i_{114.1} + (a_{-2} + a_{-1} + a_{+1} + a_{+2})i^{(c)}_{114.1} - b_{-1}i^{(c)}_{115.1} - c_{-2}i^{(c)}_{116.1}
\end{aligned}
$$

$$\Longleftrightarrow$$

$$i_{114.1} = (1 - a_{-2} - a_{-1} - a_{+1} - a_{+2})i^{(c)}_{114.1} + b_{-1}i^{(c)}_{115.1} + c_{-2}i^{(c)}_{116.1}. \tag{3.1}$$

Similar transformations with intensities of iTRAQ$^{\text{TM}}$ reporters 114.1 ... 117.1 yield the following system of linear equations comprising four equations and four variables.

$$
\begin{aligned}
i_{114.1} &= (1 - a_{-2} - a_{-1} - a_{+1} - a_{+2})i^{(c)}_{114.1} + b_{-1}i^{(c)}_{115.1} + c_{-2}i^{(c)}_{116.1} \\
i_{115.1} &= a_{+1}i^{(c)}_{114.1} + (1 - b_{-2} - b_{-1} - b_{+1} - b_{+2})i^{(c)}_{115.1} + c_{-1}i^{(c)}_{116.1} + d_{-2}i^{(c)}_{117.1} \\
i_{116.1} &= a_{+2}i^{(c)}_{114.1} + b_{+1}i^{(c)}_{115.1}(1 - c_{-2} - c_{-1} - c_{+1} - c_{+2})i^{(c)}_{116.1} + d_{-1}i^{(c)}_{117.1} \\
i_{117.1} &= b_{+2}i^{(c)}_{115.1} + c_{+1}i^{(c)}_{116.1} + (1 - d_{-2} - d_{-1} - d_{+1} - d_{+2})i^{(c)}_{117.1}
\end{aligned} \tag{3.2}
$$

3. Normalisation

Normalisation of samples is done by comparing the trimmed mean values of all iTRAQ$^{\text{TM}}$ intensities of every sample and adjusting the intensities of all samples with high peptide concentration to those of the lowest concentration by multiplicative correction. For the calculation of the normalisation factor the mean value was calculated after discarding the upper and lower 20%

of the sample's intensities (trimmed mean). In contrast to mean value or median, the trimmed mean is nearly untouched by heavy regulations of single peptides as well as regulation of significant amounts of the samples.

4. Logarithmic transformation

   The last preprocessing step is logarithmic transformation of signal intensities.

3.2 iTRAQ^{TM} specific Noise

Routinely and in accordance with Hu *et al.* (2006), Lin *et al.* (2006) and Boehm *et al.* (2007) an ion intensity dependent accuracy of regulatory data is observed. This effect can be representatively investigated and visualised by using the following test system: a protein sample is digested, split and the resulting peptides are labelled differentially with two different iTRAQ^{TM} tags (e.g. 115.1 and 117.1) before both fractions are re-combined in a 1:1 ratio and analysed by LC-MS/MS. After preprocessing (section 3.1) logarithmic peptide ratios are derived and plotted against the mean of the logarithmic reporter intensities 115.1 and 117.1 as presented in Figure 3.2. Calculation of ratios is performed by taking the logarithm of the reporter intensity quotient $\frac{115.1}{117.1}$. The peptide expression accuracy is iTRAQ^{TM} reporter ion intensity dependent. Decreasing iTRAQ^{TM} reporter ion intensities coincide with increased deviations from the expected 1:1 ratios.

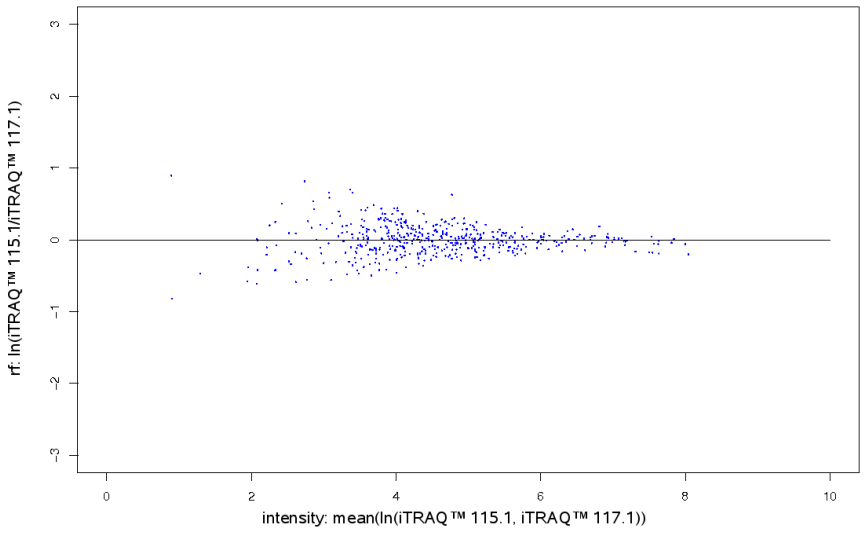


Figure 3.2: Intensity dependent noise of iTRAQ^{TM} data: decreasing iTRAQ^{TM}reporter ion intensities coincide with increased deviations from the expected 1:1 ratios. Logarithmic peptide ratios are plotted against the mean of the logarithmic reporter intensities 115.1 and 117.1.

3.3 Noise Model

Analysis of an unregulated sample – where all peptides are expected to be unregulated and therefore all ratios are expected to be 1 – produces strongly deviating regulations as shown in Figure 3.2. The quality of the derived regulatory information depends on the signal intensity. Less reliable regulation factors originating from low signal intensities are to be identified and for subsequent interpretation of biological results they are to be weighted less than regulatory information derived from high intensities. For performing this task a model is necessary that returns the possible deviation of regulation factors depending on the measured intensities.

3.3.1 Modelling

Available data Each intensity is measured several times. In the ideal case without noise, all measurements should be identical. Since noise cannot be ruled out, it is impossible to know the true intensities. The intensity range depends on properties of the analysed peptides as well as the used device.

Model Assumptions The noise follows a log-normal distribution and its variance depends on the (true, unknown) intensity. It does not seem appropriate to assume a normal distribution of the noise directly, since intensities are always non-negative. Since calculations are much easier with normal distributions, in most cases logarithms of data are used for further calculations. For the raw data and random variables associated with them the letters x and X are used, respectively, for the transformed data and their associated random variables the letters y and Y, respectively.

The general problem to be solved is as follows. A data set of the form $\left(y_1^{(1)}, \ldots, y_1^{(k_1)}, \ldots, y_n^{(1)}, \ldots, y_n^{(k_n)}\right)$ is given. $y_i^{(1)}, \ldots, y_i^{(k_i)}$ represent k_i noisy measurements of the same (logarithmic) unknown intensity μ_i.

It is assumed that the subsample $y_i^{(1)}, \ldots, y_i^{(k_i)}$ originates from independent samples of a normal distribution with unknown mean μ_i and unknown variance σ_i. From experiments it is known that the variances follow a certain tendency. Small intensities are less reliable (more noisy) than larger ones. In order to take this into account, it is assumed that

$$\sigma(\mu) \;=\; a + re^{-\lambda\mu} \tag{3.3}$$

with $a, r, \lambda \geq 0$.

a represents the absolute noise in the measurement that is always present. r specifies the amount of relative noise depending on the (logarithmic) intensity μ.

λ determines how fast the relative noise decreases with increasing (logarithmic) intensity.

3.3.2 Parameter Estimation

Parameter estimation is part of inferential statistics which aims on drawing conclusions about a population based on a sample. Maximum likelihood estimation (MLE) is a popular statistical method – with nice asymptotic properties – for the estimation of unknown parameters of a probabilistic model based on a set of observed data. The maximum likelihood estimator tries to find values for the unknown parameter to make the observation as probable as possible.

3.3.2.1 Principle of Maximum Likelihood Estimation

For a fixed set of observed data $(x_1, \dots x_n)$ and underlying probability model, the maximum likelihood estimation determines the values of the model parameters θ that make the data "most likely". A small example illustrates the approach with normally distributed data.

Assumptions:

1. X is a random variable with a random sample $x_1, ..., x_n$ following a normal distribution $\mathcal{N}(\mu, \sigma^2)$.

2. θ is the unknown parameter or parameter vector that is to be estimated.

The likelihood function equals the product of the densities corresponding to the observed values:

$$L(\theta) = \prod_{i}^{n} f(x_i \mid \theta) \tag{3.4}$$

Values of θ maximising this function are called "maximum likelihood estimator".

For the normal distribution $\mathcal{N}(\mu, \sigma^2)$ which has the probability density function (pdf)

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{3.5}$$

the likelihood is

$$L(\theta) = L(x_1, \dots, x_n \mid \theta) = \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}} = \frac{1}{(2\pi\sigma^2)^{\frac{n}{2}}} e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2}. \tag{3.6}$$

The method of maximum likelihood estimates θ by finding the value(s) of θ that maximise(s) $L(\theta)$ and therefore, that return(s) the highest probability for the random sample $x_1, \ldots, x_n$. The maximisation of the likelihood is usually carried out by finding the root of the derivative of the likelihood. The likelihood consists of multiplications and therefore, differentiation is a challenging task. The maximisation of the log-likelihood is equivalent to the maximisation of the likelihood itself. Since differentiation of the log-likelihood – consisting of summation instead of multiplication – is less complex, the log-likelihood is considered.

$$ln(L(x_1, \ldots, x_n \mid \theta)) = -\frac{n}{2} ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (x_i - \mu)^2 \tag{3.7}$$

It is not always possible to find an analytical solution for the root of the derivative for the maximum likelihood estimator. In such cases other strategies from optimisation theory are to be applied to find the maximum likelihood estimator.

3.3.2.2 Maximum Likelihood Estimation of a, r and λ

The aim is to estimate the parameters a, r and λ based on the sample $\left(y_1^{(1)}, \ldots, y_1^{(k_1)}, \ldots, y_n^{(1)}, \ldots, y_n^{(k_n)}\right)$. Unfortunately, this requires that μ_i is estimated for each subsample $y_i^{(1)}, \ldots, y_i^{(k_i)}$. It is known that the values in the subsample are noisy measurements of the same intensity μ_i, but μ_i itself is unknown.

According to the principle of maximum likelihood the parameter estimation is performed by maximising the likelihood

$$L\left(y_1^{(1)}, \ldots, y_1^{(k_1)}, \ldots, y_n^{(1)}, \ldots, y_n^{(k_n)} | a, r, \lambda\right) =$$

$$\prod_{i=1}^{n} \prod_{j=1}^{k_i} \frac{1}{(a + re^{-\lambda\mu_i})\sqrt{2\pi}} \exp\left(-\frac{(y_i^{(j)} - \mu_i)^2}{2(a + re^{-\lambda\mu_i})^2}\right). \tag{3.8}$$

The factors are the densities of normal distributions with mean μ_i and deviation $\sigma(\mu_i) = a + re^{-\lambda\mu_i}$.

The maximisation of L does not only involve the determination of the parameters a, r and λ, but also the estimation of the μ_i values. Assuming the parameters a, r and λ to be fixed at the moment, the μ_i-values can be optimised independently. This means the log-likelihoods

$$\tilde{L}_i = \sum_{j=1}^{k_i} \left(-\ln(\sqrt{2\pi}) - \ln(h(\mu_i)) - \frac{(y_i^{(j)} - \mu_i)^2}{2(h(\mu_i))^2}\right) \tag{3.9}$$

where $h(\mu_i) = a + re^{-\lambda\mu_i}$ are to be maximised. In order to maximise $\tilde{L}_i$ it is necessary that

$$\frac{d\tilde{L}_i}{d\mu_i} = \sum_{j=1}^{k_i} \left(-\frac{h'(\mu_i;\theta)}{h(\mu_i;\theta)} + \frac{(x_i^{(j)} - \mu_i)(h(\mu_i;\theta) + (x_i^{(j)} - \mu_i)h'(\mu_i;\theta))}{h^3(\mu_i;\theta)} \right) = 0 \tag{3.10}$$

holds. With $h'(\mu_i) = -\lambda r e^{-\lambda\mu_i}$ and multiplying (3.10) by $(a + re^{-\lambda\mu_i})^3$

$$\sum_{j=1}^{k_i} \left((\lambda r e^{-\lambda\mu_i})(a + re^{-\lambda\mu_i})^2 + (x_i^{(j)} - \mu_i)((a + re^{\lambda\mu_i}) + (x_i^{(j)} - \mu_i)(-\lambda r e^{-\lambda\mu}) \right) = 0 \tag{3.11}$$

is obtained. Solving (3.11) for μ_i yields the maximum likelihood estimation for μ_i, assuming the parameters a, r and λ to be fixed. This is done numerically by a simple bisection strategy. As one boundary for bisection, the mean value of the $y_i^{(j)}$ is chosen. The second one is determined by systematically searching left and right from this value until the sign of (3.11) changes. The optimisation of the parameters a, r and λ is carried out by an evolution strategy (Bäck, 1996).

Evolution Strategy Evolution strategies are search and optimisation methods using mutation, recombination, and selection applied to a population of individuals containing candidate solutions in order to evolve iteratively better and better solutions. The principle of evolution contains

1. initialisation of a parent population

2. generation of an offspring population by recombination and variation of existing elements ("mutation")

3. selection of a new parent population from either the offspring population or the union of offspring and parent population.

The mutation and selection steps are repeated until the termination criterion is fulfilled.

For the optimisation of the parameters a, r and λ the applied evolution strategy has an adaptive mutation rate and population size = 10, offspring size = 25. The mutation of the parameters (offsprings) is given by adding a normally distributed random number which is influenced by the mutation rate. The algorithm terminates after 100 generations or if 20 generations could not achieve an improvement of fitness. The fitness of a parameter combination (a, r, λ) is given by (3.8), where the μ_i are determined as described above based on solving (3.11).

3.3.3 Training

The parameters a, r, and λ directly depend on the type of the mass spectrometer in general as well as on data acquisition settings. Parameter estimation for every analysis dataset individually causes problems concerning (i) the low number of samples, (ii) optionally comprised regulated peptides and last but not least (iii) extension of runtime for each analysis. Particularly the low number of samples – usually two samples are analysed which is not enough for statistical calculations – is an important aspect. To estimate the parameters a, r, λ a training dataset is generated. Five peptides are used to determine the intensity dependent noise. Two unmodified peptides and three phosphopeptides are included in this approach which facilitate to cover a broad dynamic reporter intensity range. Measurements are repeated 23 times (sample size $k_i = 23$ for all i) for all five peptides. By using different collision energies for the measurements, different absolute levels of iTRAQ$^{\text{TM}}$ reporter intensities are obtained. In this way 23 reporter intensity variations at 24 different intensity levels resulting in 552 individual reporter intensities are repeatedly measured.

From this dataset the noise parameters were estimated for the used instruments and postprocessing settings (compare section 2.4) as follows: $a = 0.0103$, $r = 0.9908$ and $\lambda = 0.4751$.

Table 3.2 itemises the sample ID, the peptide sequence as well as the applied collision energy for every of the 24 subsets of the training dataset. Tables 3.3 and 3.4 show measured intensities, mean intensity and standard deviation of every subset.

Table 3.2: Subset ID, peptide sequence and applied collision energy of the 24 subsets of the training dataset. Phopshorylated amino acids are coloured.

Subset ID	Sequence	collision energy [eV]
Z4298A	FVLDDQYTSSTGTKFPVK	34.5
Z4299	FVLDDQYTSSTGTKFPVK	30
Z4299A	FVLDDQYTSSTGTKFPVK	25
Z4299D	FVLDDQYTSSTGTKFPVK	35
Z4300A	SSTVTEAPIAVVTSR	30
Z4300B	SSTVTEAPIAVVTSR	40
Z4300n	SSTVTEAPIAVVTSR	35
Z4309	HERPAGPGTPPPDSGPLAK	25
Z4309C	HERPAGPGTPPPDSGPLAK	22
Z4309A	HERPAGPGTPPPDSGPLAK	28
Z4309B	HERPAGPGTPPPDSGPLAK	30
Z2232	VSDFGLTK	30
Z2233	VSDFGLTK	28
Z2234	VSDFGLTK	26
Z2235	VSDFGLTK	25
Z2236	VSDFGLTK	22
Z2240	VSDFGLTK	12
Z2241	IADPEHDHTGFLTEYVATRWYR	37.8
Z2242	IADPEHDHTGFLTEYVATRWYR	37.8
Z2243	IADPEHDHTGFLTEYVATRWYR	37.8
Z2244	IADPEHDHTGFLTEYVATRWYR	35
Z2245	IADPEHDHTGFLTEYVATRWYR	33
Z2246	IADPEHDHTGFLTEYVATRWYR	32
Z2247	IADPEHDHTGFLTEYVATRWYR	30

Table 3.3: Measured raw intensities, mean value and standard deviation of the subsets z2232, z2233, z2235, z2236, z2241, z2242, z2243, z2244, z2245, z2246 and z2247 of the training dataset.

Subset ID	z2232	z2233	z2234	z2235	z2236	z2241	z2242	z2243	z2244	z2245	z2246	z2247
	14805	11934	8618	5792	3430	1492	1358	1243	768	550	434	298
	14583	12113	8418	5840	3400	1355	1394	1219	812	555	431	282
	14523	11859	8739	5547	3256	1318	1402	1189	760	542	378	260
	14498	12335	9049	5633	3347	1324	1325	1252	726	598	415	289
	14654	12424	8871	5774	3475	1367	1310	1220	732	488	422	260
	14536	12182	8913	5750	3557	1409	1235	1217	773	529	469	276
	14724	12084	8824	5813	3360	1347	1262	1192	741	527	456	308
	14829	12254	9036	5763	3483	1319	1274	1290	752	522	425	327
	14802	12329	8755	5823	3421	1352	1325	1244	737	532	446	286
	14916	12036	8836	5875	3315	1373	1274	1246	739	547	418	295
	14575	12376	8800	5881	3433	1333	1313	1205	829	497	458	289
	14785	12001	8960	5931	3455	1379	1371	1239	804	534	452	267
	14830	12133	8795	5927	3460	1361	1364	1271	762	508	466	295
	14498	12020	8911	5835	3479	1314	1252	1201	792	521	438	316
	14422	11870	8935	5722	3461	1357	1306	1241	738	510	457	323
	14302	11613	8931	5765	3605	1404	1273	1229	717	516	466	282
	14568	11232	8924	5909	3673	1334	1251	1194	777	509	437	249
	14577	11218	8730	6053	3514	1285	1265	1171	780	470	387	292
	14652	11246	8728	5843	3496	1423	1269	1089	769	559	450	324
	14426	11404	8850	5827	3586	1402	1267	1150	736	538	433	285
	14601	11356	8692	5855	3579	1307	1252	1092	781	530	431	279
	14288	11062	8662	5919	3618	1430	1239	1155	813	542	420	289
	14514	11487	8547	5737	3705	1331	1224	1143	732	531	440	333
$\bar{x}$	**14604.7**	**11850.78**	**8805.39**	**5818**	**3482.96**	**1361.57**	**1295.87**	**1204**	**763.91**	**528.48**	**436.04**	**291.48**
sd	**169.43**	**428.53**	**153.37**	**105.53**	**112.05**	**48.02**	**52.41**	**51.93**	**31.16**	**26.31**	**23.31**	**22.39**

Table 3.4: Measured raw intensities, mean value and standard deviation of the subsets z2240, Z4298A, Z4299, Z4299A, Z4299D, Z4300A, Z4300B, Z4300n, Z4309, Z4309C, Z4309A and Z4309B of the training dataset.

Subset ID	z2240	Z4298A	Z4299	Z4299A	Z4299D	Z4300A	Z4300B	Z4300n	Z4309	Z4309C	Z4309A	Z4309B
	107	89	259	61	634	58	185	132	107	33	205	313
	93	91	254	49	707	45	213	121	83	36	212	285
	106	87	262	49	763	52	226	111	77	36	193	346
	113	74	207	49	691	37	240	110	86	44	214	323
	118	75	263	72	698	29	297	121	112	36	194	295
	108	89	236	67	674	40	281	112	98	36	228	310
	123	79	248	50	675	43	234	126	96	41	180	312
	122	104	297	71	631	46	236	142	92	46	205	321
	108	93	300	72	659	31	215	143	103	28	206	333
	123	101	284	45	729	36	237	111	84	42	201	332
	99	85	264	40	680	50	216	106	78	32	204	350
	109	97	322	51	681	38	224	108	103	36	203	322
	99	84	310	52	657	46	243	120	90	46	203	302
	119	81	322	43	612	47	228	99	80	32	232	351
	109	89	282	47	646	40	274	105	104	39	233	313
	119	86	253	52	642	27	256	133	109	43	191	301
	129	96	298	55	556	38	268	135	105	25	231	305
	95	92	205	50	617	38	238	128	104	29	241	311
	97	88	198	75	531	44	249	137	82	45	205	329
	116	72	255	46	658	52	231	103	110	35	237	340
	103	82	224	52	601	43	243	112	104	34	237	325
	99	85	281	47	679	32	234	107	93	32	205	306
	95	74	299	62	643	45	234	129	85	46	234	295
$\bar{x}$	**109.09**	**86.65**	**266.22**	**54.65**	**654.96**	**41.61**	**239.22**	**119.61**	**95**	**37.04**	**212.78**	**318.26**
sd	**10.55**	**8.54**	**36.04**	**10.31**	**51.55**	**7.77**	**24.29**	**13.27**	**11.33**	**6.16**	**17.52**	**18.12**

3.3.4 Validation of the Noise Model

Several assumptions were made for the development of the noise model, especially (i) log-normally distributed intensities and (ii) the definition of the standard deviation by $\sigma(\mu) = a + re^{\lambda\mu}$. These assumptions are to be verified. This can be performed by application of statistical tests to the underlying data as well as simulation of data corresponding to the proposed model. Furthermore, the so-called "95% Interval" can be used for the comparison of the expected deviation according to the model and the deviation of an unregulated sample.

3.3.4.1 Verification of Assumptions

By the means of several statistical techniques the verification of the made assumptions is realised. Primarily, statistical tests and simulation of data generated by the proposed model and the estimated parameters (3.3.3) are applied.

Testing for Normal Distribution

For testing whether the noise follows a log-normal distribution, the logarithmic intensities of each subset of the training dataset is tested for following a normal distribution. Therefore, in the following logarithmic intensities are regarded. First of all, two statistical tests checking the goodness-of-fit are applied and additionally quantile-quantile plots are generated for each subset.

Statistical hypothesis tests calculate the probability of an observation assuming the null hypothesis is valid. This involves the test of a null hypothesis H_0 against an alternative hypothesis H_1. For example, the null hypothesis could be *"the observed data follow a normal distribution"*. In that case, the alternative hypothesis would be *"the observed data do not follow a normal distribution"*. According to the applied test statistic the probability of the observation is computed assuming that the null hypothesis is valid. If this value, also known as p-value, is less than the defined significance level (e.g. 5% and 0.05, respectively), the null hypothesis is rejected in behalf of the alternative hypothesis. Otherwise, the null hypothesis can not be rejected, which is no proof of its validity. For details see for example W. J. Ewens (2002).

Both Shapiro-Wilk-Test and Kolmogorov-Smirnov-Test are goodness-of-fit-tests that are able to check whether the assumption of a sample being normally distributed has to be rejected or not.

Shapiro-Wilk-Test The Shapiro-Wilk-Test (Shapiro and Wilk, 1965) checks the assumption that a random sample was drawn from a normal distribution. According to L. Sachs (2006) the Shapiro-Wilk-Test has the highest power (probability

of rejecting a false null hypothesis) in comparison with other statistical tests. It is most reliable for small sample sizes ($n < 50$).

The results achieved by application of Shapiro-Wilk-Test to the subsets of the training dataset are in almost all cases the same: The null hypothesis *"the intensities of the subsets follow a normal distribution"* can not be rejected. Just subset Z2233 achieved a p-value < 0.05 and thus is not regarded to be normally distributed.

Kolmogorov-Smirnov-Test The Kolmogorov-Smirnov-Test (K-S-Test) – developed in the 1930s – allows the comparison of an assumed distribution to some other known distribution. Unlike Shapiro-Wilk-Test it is possible to test for different distributions, for example the normal distribution. If the assumed distribution is not completely known, e.g. if the expectation value and/or the variance are estimated, the results of the K-S-Test are inaccurate (L. Sachs, 2006).

K-S-Test returned that the intensities of no subsets follow a normal distribution. Regarding the comments of L. Sachs (2006) concerning the accuracy of this test when expectation value and variance are estimated – as in the case of the analysed subsets – these results are weighted less than the results of Shapiro-Wilks-Test.

Results

The calculated p-values of all subsets obtained by Shapiro-Wilk-Test and K-S-Test are summarised in Table 3.5.

Table 3.5: Comparison of the resulting p-values of Shapiro-Wilk-Test and Kolmogorov-Smirnov-Test. Regarding Shapiro-Wilk-Test the null hypothesis *"the intensities of the subsets follow a normal distribution"* can not be rejected, whereas K-S-Test determines the opposite.

Subset ID	Shapiro-Wilk-Test	Kolmogorov-Smirnov-Test
Z4298A	0.76	$1.3e^{-4}$
Z4299	0.19	$3.4e^{-4}$
Z4299A	0.05	$2e^{-5}$
Z4299D	0.38	$1.8e^{-4}$
Z4300A	0.69	$3.7e^{-4}$
Z4300B	0.40	$3.7e^{-4}$
Z4300n	0.23	$5e^{-5}$
Z4309	0.06	$6.5e^{-5}$
Z4309C	0.24	$2.73e^{-3}$
Z4309A	0.06	$8e^{-7}$
Z4309B	0.81	$7e^{-6}$
Z2232	0.53	$8e^{-7}$
Z2233	0.04	$2e^{-7}$
Z2234	0.51	$2.4e^{-6}$
Z2235	0.51	$2e^{-7}$
Z2236	0.98	$2e^{-5}$
Z2240	0.31	$1.75e^{-3}$
Z2241	0.37	$3.5e^{-7}$
Z2242	0.07	$8e^{-7}$
Z2243	0.13	$2e^{-6}$
Z2244	0.32	$5.5e^{-5}$
Z2245	0.75	$7e^{-5}$
Z2246	0.78	$1.6e^{-4}$
Z2247	0.59	$3.8e^{-4}$

Quantile-Quantile Plots Since the results of the applied goodness-of-fit-tests are inconsistent, an additional test for checking the assumption is performed. A quantile-quantile plot (Q-Q plot) is a graphical method for diagnosing differences between the probability distribution of a statistical population (from which a random sample has been taken) and a comparison distribution. The quantiles of the sample are plotted against the theoretical quantiles of the comparison distribution where a q-quantile is given by $x_q = P(X \leq x_q)$. If the n points (n = sample size) approximate a straight line, the population distribution is the same as the comparison distribution. In the case of using a normal distribution as comparison distribution the plots are called "normal Q-Q plots".

Figures 3.3 and 3.4 show normal Q-Q plots for data following a normal distribution (Figure 3.3) and for data following a uniform distribution (3.4). In contrast to uniformly distributed data, the normally distributed data approximately fit a line.

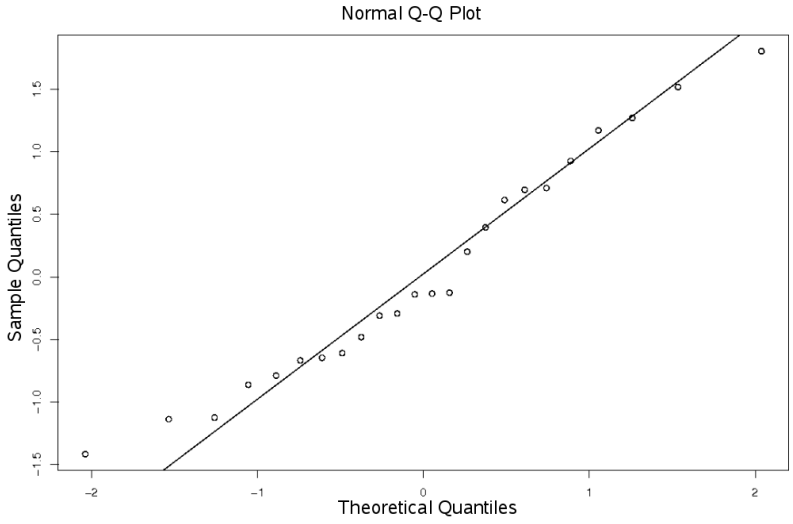


Figure 3.3: Examples of a normal Q-Q plot for data following a normal distribution. The sample quantiles are plotted against the theoretical quantiles approximating a line.

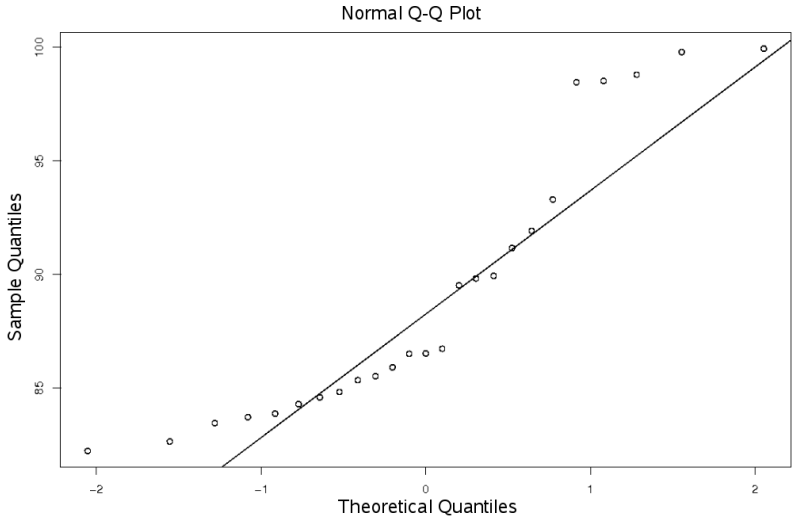


Figure 3.4: Example of a normal Q-Q plot for data following a uniform distribution. The sample quantiles are plotted against the theoretical quantiles not approximating a line.

The following figures (Figure 3.5 – Figure 3.7) show the normal Q-Q plots for all subsets of the training dataset. Most of them confirm or – at least – are not contrary to the assumption that the logarithmic intensities are following a normal distribution and consequently, that the intensities are log-normally distributed.

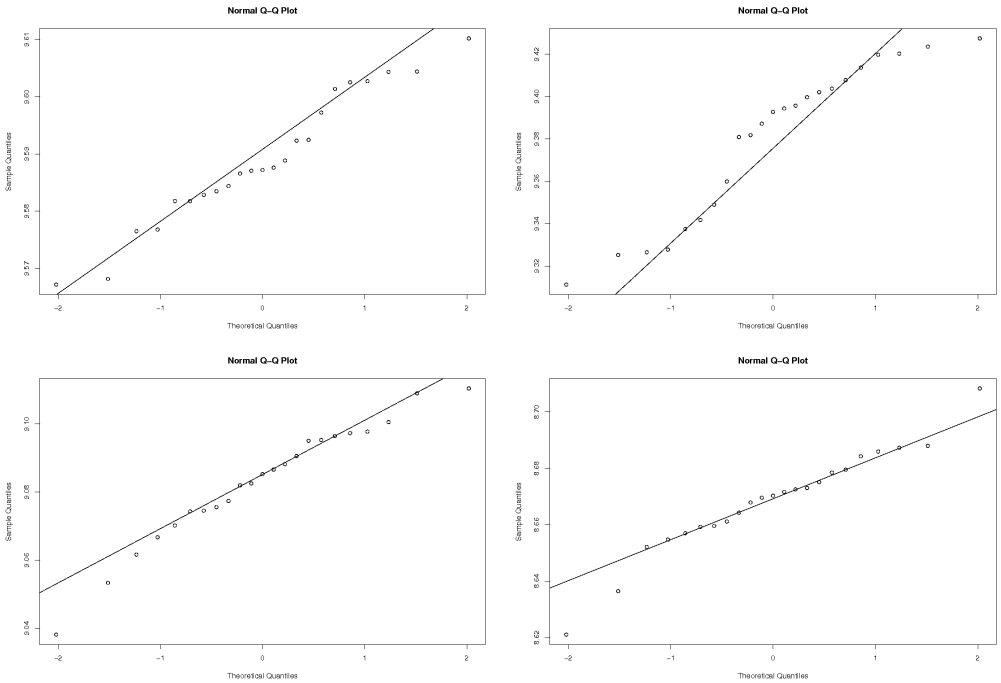


Figure 3.5: Normal Q-Q plots of the subsets Z2232, Z2233, Z2234, and Z2235. The sample quantiles are plotted against the theoretical quantiles.

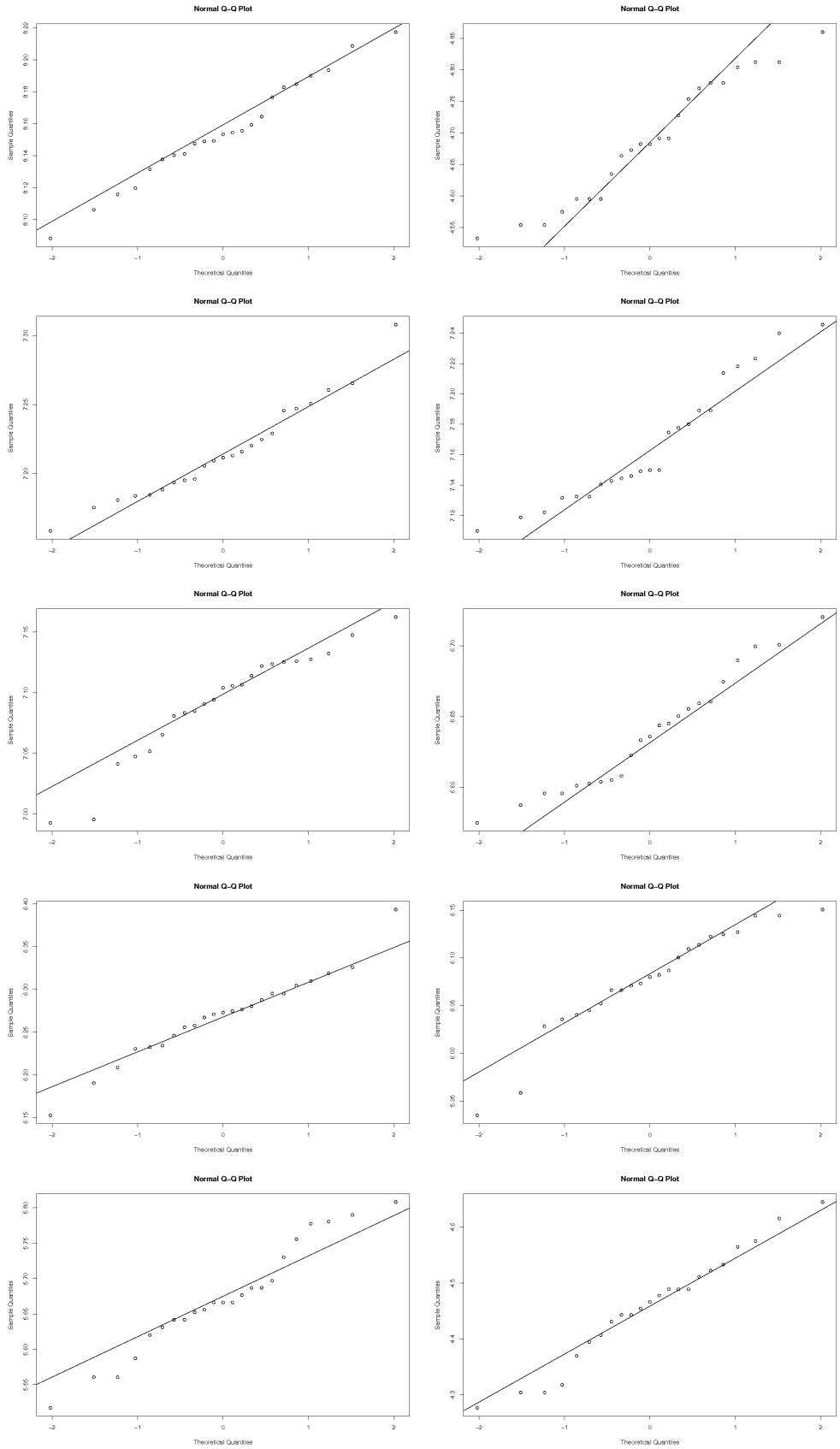


Figure 3.6: Normal Q-Q plots of the subsets Z2236, Z2240, Z2241, Z2242, Z2243, Z2244 Z2234, Z2246, Z2247 and Z4298A. The sample quantiles are plotted against the theoretical quantiles.

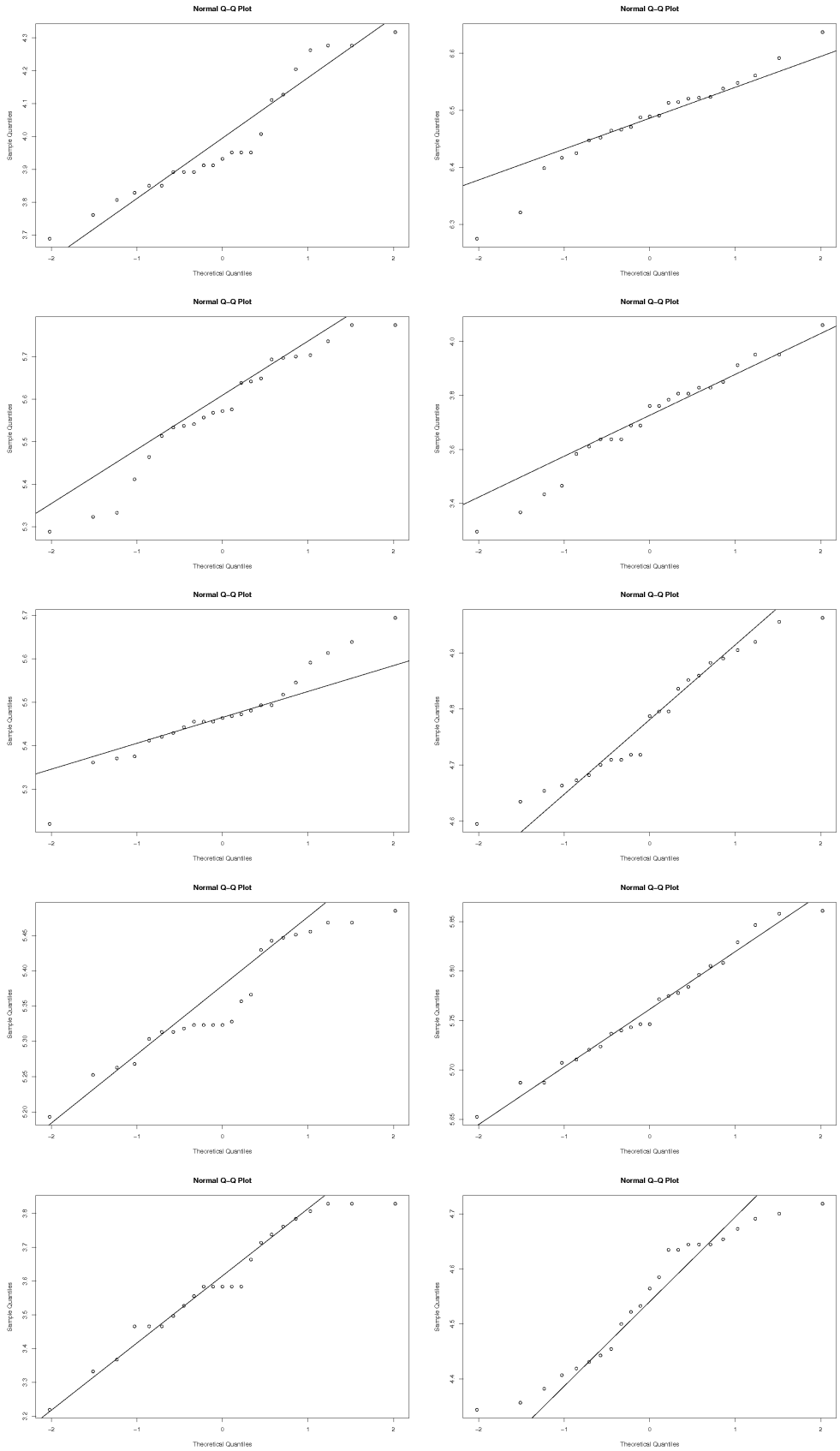


Figure 3.7: Normal Q-Q plots of the subsets Z4299A, Z4299D, Z4299, Z4300A, Z4300B, Z4300n, Z4309A, Z4309B, Z4309C and Z4309. The sample quantiles are plotted against the theoretical quantiles.

Simulations

The applied simulations provide a basis for checking the proposed noise model. Especially the assumption of the standard deviation $\sigma(\mu) = a + re^{-\lambda\mu}$ is verified in this way. Both simulations have in common that the estimated parameter values are used for the generation of datasets and the results are compared with results achieved by analysis of the test dataset.

Comparison of Variances The variances of the subsets of the training dataset and a simulated dataset generated by the noise model are compared to each other. On the one hand, the mean value and the variance of every subset are calculated (Table 3.6, columns "mean subset" and "var subset"). On the other hand, corresponding to every subset 100 normally distributed random numbers are generated with a mean value equal to the mean value of the regarded subset. From this dataset the mean value and the variance are determined (Table 3.6, columns "mean model" and "var model") with var model $= \sigma^2(\mu) = (a+re^{-\lambda\mu})^2$, $a = 0.0103$, $r = 0.9908$ and $\lambda = 0.4751$ (as estimated, compare section 3.3.2). The results are given in Table 3.6. In summary, the deviations of mean values and variances of the training dataset subsets and the simulated datasets are smallish (mean deviation $= 0.0018$) and therefore, serve as a first confirmation of the assumption $\sigma^2(\mu) = (a + re^{-\lambda\mu})^2$.

Table 3.6: Comparison of variances of the training dataset's subsets and simulated subsets with μ = mean value of subset and $\sigma^2(\mu) = (a + re^{-\lambda\mu})^2$. The last column refers to the differences.

Sample ID	mean subset	var subset	mean model	var model	|var subset - var model|
Z4298A	4.4572	0.0168	4.4358	0.0167	**0.0001**
Z4299	5.5751	0.0065	5.5733	0.0078	**0.0013**
Z4299A	3.9850	0.0254	3.9955	0.0280	**0.0026**
Z4299D	6.4815	0.0031	6.4912	0.0028	**0.0003**
Z4300A	3.7109	0.0325	3.7311	0.0372	**0.0047**
Z4300B	5.4725	0.0070	5.4810	0.0065	**0.0005**
Z4300n	4.7784	0.0127	4.7675	0.0119	**0.0008**
Z4309	4.5469	0.0155	4.5360	0.0149	**0.0006**
Z4309C	3.5984	0.0359	3.5746	0.0332	**0.0027**
Z4309A	5.3570	0.0076	5.3573	0.0100	**0.024**
Z4309B	5.7613	0.0055	5.7634	0.0057	**0.0002**
Z2232	9.5890	0.0004	9.5898	0.0004	**0.0000**
Z2233	9.3795	0.0005	9.3783	0.0004	**0.0001**
Z2234	9.0830	0.0006	9.0830	0.0007	**0.0001**
Z2235	8.6686	0.0007	8.6662	0.0007	**0.0000**
Z2236	8.1551	0.0010	8.1567	0.0011	**0.0001**
Z2240	4.6877	0.0137	4.6709	0.0130	**0.0007**
Z2241	7.2158	0.0018	7.2182	0.0017	**0.0001**
Z2242	7.1662	0.0019	7.1709	0.0020	**0.0001**
Z2243	7.0925	0.0020	7.0947	0.0022	**0.0002**
Z2244	6.6377	0.0028	6.6345	0.0028	**0.0000**
Z2245	6.2688	0.0037	6.2634	0.0039	**0.0002**
Z2246	6.0763	0.0043	6.0767	0.0040	**0.0003**
Z2247	5.6722	0.0060	5.6766	0.0066	**0.0006**

Re-estimating the Standard Deviation Based on the estimated parameters $a = 0.0103$, $r = 0.9908$ and $\lambda = 0.4751$ an analysis dataset is simulated. From the simulated dataset the parameters a, r, λ are re-estimated. For this purpose, 200 logarithmic intensity pairs are created by generating two normally distributed random numbers corresponding to intensities between 15 and 4000 before taking the logarithm. In the used programming language (Java[1]), generation of normally distributed random numbers is only possible for the standard normal distribution $\mathcal{N}(0, 1)$. Therefore, a transformation of the random numbers following $\mathcal{N}(0, 1)$ towards $\mathcal{N}(\mu, \sigma^2)$ with $\sigma(\mu) = a+re^{-\lambda\mu}$ is necessary. Transformations from $\mathcal{N}(\mu, \sigma^2)$ to $\mathcal{N}(0, 1)$ are given by

$$Z = \frac{X - \mu}{\sigma} \tag{3.12}$$

[1]http://java.sun.com/

where Z is the transformed value of X which is the $\mathcal{N}(\mu, \sigma^2)$ distributed measured value. Consequently, transformation from $\mathcal{N}(0, 1)$ to $\mathcal{N}(\mu, \sigma^2)$ is achieved by

$$X = Z\sigma + \mu. \tag{3.13}$$

From these simulated intensity pairs the parameters a, r, λ are re-estimated resulting in $a = 0.0117$, $r = 0.9857$ and $\lambda = 0.5348$. Since these values are similar to those, used for the generation of the simulated dataset, the proposed model is empirically consistent.

3.3.4.2 95% Interval

In order to find out whether the trained model and the intensity deviation of unregulated samples fit together, a so-called 95% interval is determined. The idea is to calculate the 95% interval for each normally distributed integer logarithmic intensity as illustrated in Figure 3.8. Finally, 95% of the expression ratios of a sample should be located inside the interval and 5% outliers are accepted.

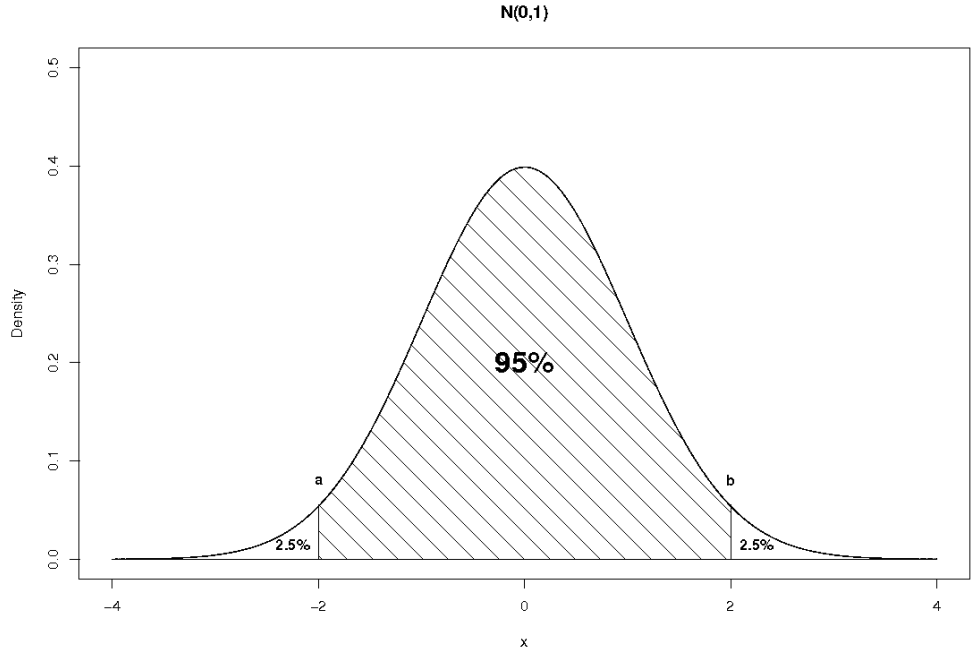


Figure 3.8: Standard normal distribution. The values a and b give the borders of the marginal 2.5% for the calculation of the 95% interval of intensities.

Therefore, the marginal borders a and b with $P(a \leq Z \leq b) = 0.95$ are to be found. This is carried out by the means of statistical standard algorithms that are available for the standard normal distribution $\mathcal{N}(0, 1)$. Assuming two identically regulated samples with n pairs of logarithmic intensities $(y_i^{(1)}, y_i^{(2)})$, $1 \leq i \leq n$,

follows $\mu_i^{(1)} = \mu_i^{(2)}$. Subtraction of the means and addition of the variances leads to

$$\begin{aligned} \mathcal{N}(\mu, \sigma^2) &= \mathcal{N}(\mu^{(1)} - \mu^{(2)}, (\sigma^{(1)})^2 + (\sigma^{(2)})^2) \\ &= \mathcal{N}(0, (a + re^{-\lambda\mu^{(1)}})^2 + (a + re^{-\lambda\mu^{(2)}})^2) \\ &= \mathcal{N}(0, 2(a + re^{-\lambda\mu})^2). \end{aligned} \tag{3.14}$$

The calculated borders are transformed from $\mathcal{N}(0, 1)$ to $\mathcal{N}(\mu, \sigma^2)$ by application of (3.13). Hence, the borders are x and $-x$, respectively, with

$$x = z\sqrt{2(a + re^{-\lambda\mu})^2} \tag{3.15}$$

where $a = 0.0103, r = 0.9908, \lambda = 0.4751$ and $\mu =$ the currently regarded logarithmic intensity.

The values for 95% interval exclusively depend on the estimated parameters. Consequently, they are the same for all datasets shown in the following paragraphs.

Training Dataset Positioning of the training dataset that was used for the estimation of the model parameters within the 95% interval serves as proof-of-concept. Pseudo-expression ratios are calculated by dividing the measured intensities of each subset by its mean value. Figure 3.9 shows 24 subsets each consisting of 23 individual intensity measurements. The curves give the 95% interval for normally distributed noisy intensities with $\mathcal{N}(0, \sqrt{2\sigma^2})$.

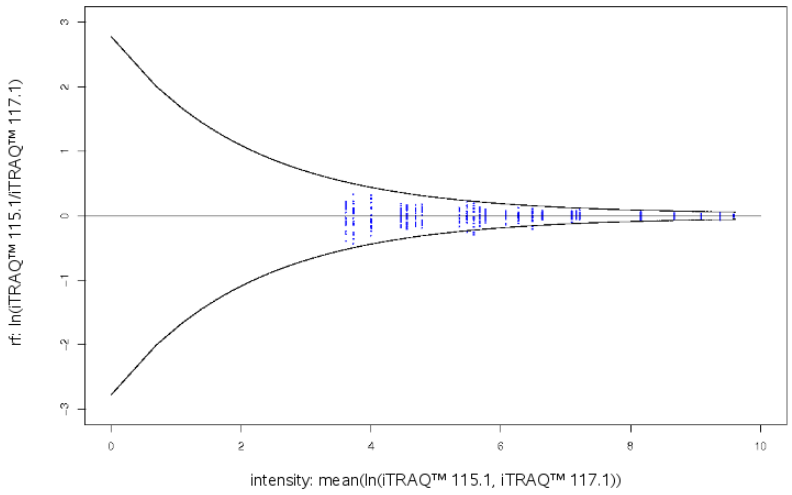


Figure 3.9: Logarithms of pseudo-expression ratios from the training dataset within the 95% interval. Each subset of the training dataset represents one intensity, pseudo-expression ratios are calculated by dividing the measured intensities by the mean value of the intensities of the corresponding subset.

Test Dataset The experimental test dataset (Figure 3.10, also compare Figure 3.2) comprises logarithmic expression ratios of 563 peptides. Tryptically peptides from six different proteins were differentially labelled with iTRAQ^{TM} 115.1 or 117.1, mixed in the ratio 1:1 and then analysed by MS. 2.84% of the peptide regulation factors are located outside the 95% interval – given by the black curves – which is calculated from the training dataset.

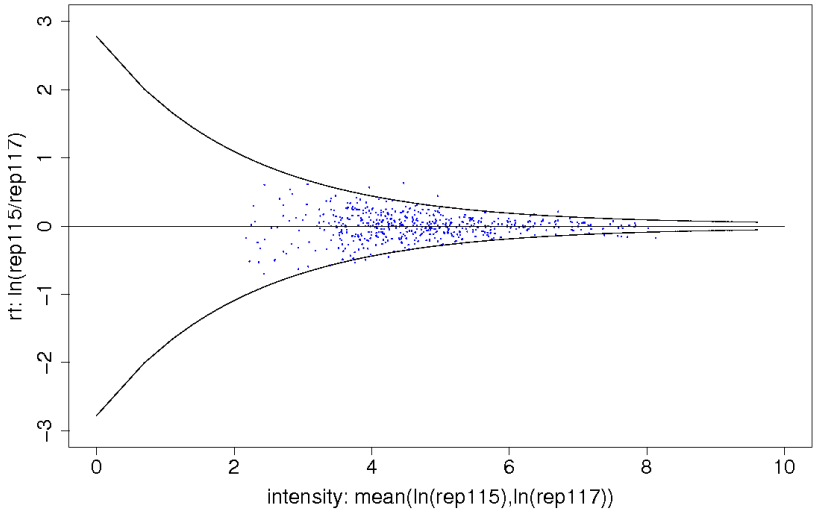


Figure 3.10: Logarithms of expression ratios from the training dataset are plotted against the logarithmic mean intensity from both differentially labelled subsamples. The black lines give the 95% interval, 2.85% of the regulation factors are located outside the interval.

Simulation Simulation of a dataset based on the model parameters that were estimated from the training dataset (Figure 3.11) shows striking similarities to the intensities detected in the experimental test dataset (Figure 3.10). Every integer intensity between 1 and 2000 (e^{μ}) was used for the generation of two noisy intensities ($sim1$ and $sim2$) according to the noise model with the parameters $a = 0.0103$, $r = 0.9908$ and $\mu = 0.4751$. Values of $ln(\frac{sim1}{sim2})$ are plotted against μ.

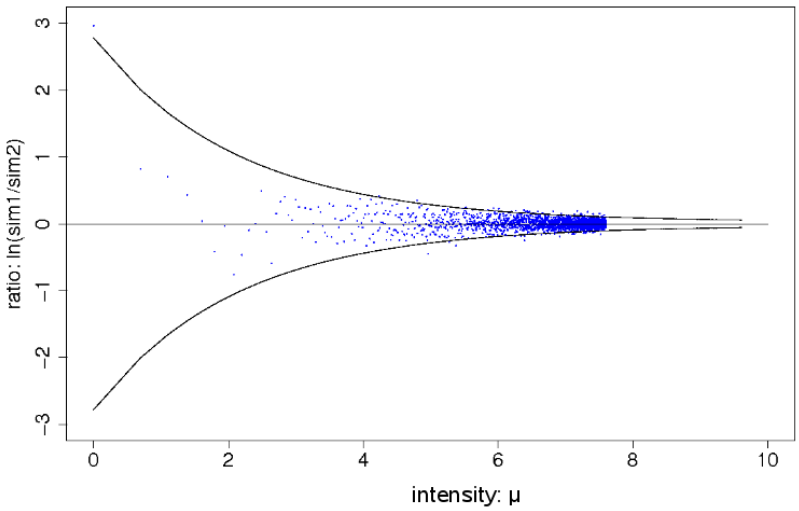


Figure 3.11: Logarithms of expression ratios from a simulated unregulated sample within the 95% interval. Every integer intensity between 1 and 2000 (e^{μ}) was used for the generation of two noisy intensities ($sim1$ and $sim2$). Values of $ln(\frac{sim1}{sim2})$ are plotted against μ.

Complex Sample Tryptically generated peptides from a complex biological sample were differentially labelled with iTRAQ$^{\mathrm{TM}}$ 115.1 or 117.1, mixed in the ratio 1:1 and then analysed by MS. 1.9% of the peptide expression ratios are located outside the 95% interval supporting the conservative and representative character of the established noise model.

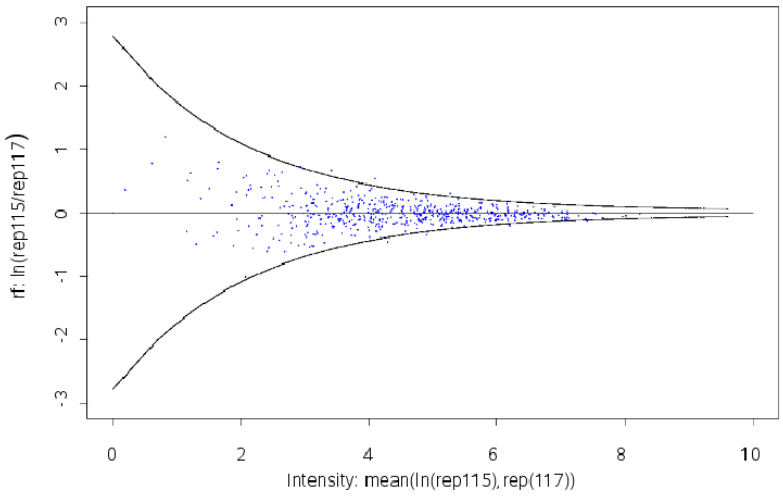


Figure 3.12: Logarithms of expression ratios from an unregulated complex biological sample are plotted against the logarithmic mean intensity from both differentially labelled subsamples. The black lines give the 95% interval, 1.90% of the regulation factors are located outside the interval.

In summary, all made assumptions

(i) log-normally distributed intensities

(ii) $\sigma(\mu) = a + re^{-\lambda\mu}$

were approved – or at least, could not be rejected – by the applied tests for the validation of the noise model.

3.3.5 Applications of the Noise Model

Besides the identification of significant regulations by finding the most probable regulation factor (chapter 4), two applications for robustness analysis of intensities and regulation could be derived from the noise model.

Error Probability of Contrary Regulation

The error probability of contrary regulation gives the probability that the measurements and the corresponding true intensities are in the same order. For example, if x_1 was measured smaller than x_2 the error probability gives the probability that also the true intensity of x_1 is smaller than that of x_2. Assuming that e^{μ_1} and e^{μ_2} are the true intensities of x_1 and x_2, respectively it is supposed that $e^{\mu_1} \geq e^{\mu_2}$, i.e. the true intensities are not in the same order as the measured intensities. This happens with probability

$$2 \cdot P(Y_{\mu_2} \geq \ln(x_2)) \cdot P(Y_{\mu_1} \leq \ln(x_1)). \tag{3.16}$$

The factor 2 reflects that the order (in time) in which x_1 and x_2 were measured is not considered. Without the factor 2 the meaning were that the larger value x_2 is measured after the smaller value x_1. μ_1 and μ_2 are to be found with $\mu_2 \leq \mu_1$ such that (3.16) is maximised. It is obvious that μ_2 should be as large as possible in order to maximise $P(Y_{\mu_2} \geq \ln(x_2))$, whereas μ_1 should be as small as possible in order to maximise $P(Y_{\mu_1} \leq \ln(x_1))$. Figure 3.13 shows three examples of choosing μ_1 and μ_2 given measurements of x_1 and x_2. In the upper example μ_1 and μ_2 have a large distance to each other, in the middle they are nearer and in the last example μ_1 and μ_2 are identical. The areas to be maximised corresponding to $P(Y_{\mu_2} \geq \ln(x_2))$ and $P(Y_{\mu_1} \leq \ln(x_1))$, respectively, are coloured. Obviously, the sum of the coloured areas is the highest if $\mu_1 = \mu_2$.

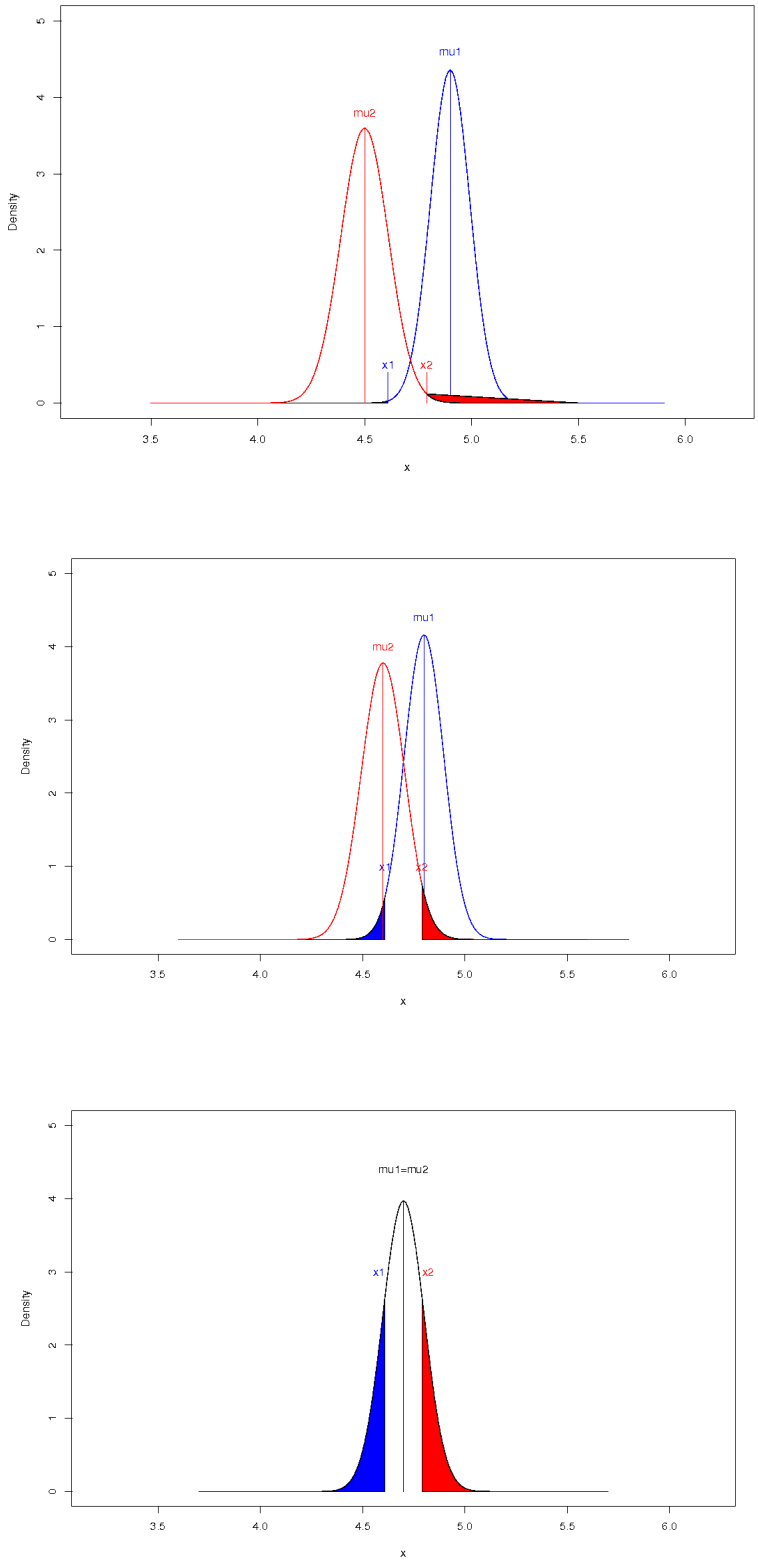


Figure 3.13: Three examples of choosing μ_1 and μ_2 given the measurements x_1 and x_2. The areas to be maximised corresponding to $P(Y_{\mu_2} \geq \ln(x_2))$ and $P(Y_{\mu_1} \leq \ln(x_1))$, respectively, are coloured. Maximum of the sum of the coloured areas is found if $\mu_1 = \mu_2$.

With the constraint $\mu_2 \leq \mu_1$, (3.16) will only achieve its maximum if $\mu_1 = \mu_2$ holds. This means, maximising (3.16) is equivalent to maximise

$$2 \cdot P(Y_\mu \geq \ln(x_2)) \cdot P(Y_\mu \leq \ln(x_1)). \tag{3.17}$$

with the parameter μ. (3.17) is equivalent to the objective function

$$g(\mu) \;=\; 2 \cdot \left(1 - \Phi\left(\frac{\ln(x_2) - \mu)}{a + re^{-\lambda\mu}}\right)\right) \cdot \Phi\left(\frac{\ln(x_1) - \mu)}{a + re^{-\lambda\mu}}\right) \tag{3.18}$$

where Φ is the cumulative distribution function of the standard normal distribution. This is done again by an evolution strategy with adaptive mutation rates.

The maximum value of $g(\mu)$ is the maximum probability that two true intensities with the reverse order of x_1 and x_2 can produce values like x_1 and x_2 but it is not the probability that the true intensities for x_1 and x_2 are in the reverse order. The computed probability, i.e. the maximum value of the function g, is the highest possible probability that two true intensities e^{μ_1} and e^{μ_2} with $\mu_2 \leq \mu_1$ can produce an intensity pair like x_1 and x_2.

Considering a dataset of statistical inferences simultaneously causes the multiple comparisons problem. Errors in inference, e.g. hypothesis tests incorrectly rejecting the null hypothesis, are more likely when the dataset is considered as a whole. In order to avoid this problem the significance level of the resulting values is adapted by Bonferroni correction to $\frac{1}{n}$ of the significance level of each single comparison where n is the number of peptides.

Intensity Interval

The intensity interval gives the range of the true intensity corresponding to a measurement x according to a defined significance level. A value α (e.g. 0.05) must be specified determining the probability to which it is acceptable that the log-normal random variable $X_{\mu_{\max}}$ for the true intensity $e^{\mu_{\max}}$ will produce a value less than or equal to x if $e^{\mu_{\max}} > x$ holds. $e^{\mu_{\max}}$ will be the upper bound for x. Similarly, for the lower bound $e^{\mu_{\min}} < x$, α is the probability that the log-normal random variable $X_{\mu_{\min}}$ produces a value larger than or equal to x. $Y_{\mu_{\max}} = \ln(X_{\mu_{\max}})$ and $Y_{\mu_{\min}} = \ln(X_{\mu_{\min}})$ denote the corresponding normal distributions. Figure 3.14 illustrates this context: $Y_{\mu_{\max}}$ and $Y_{\mu_{\min}}$ are able to produce the intensity $ln(x)$ with certain probabilities. $\mu_{\max}$ and $\mu_{\min}$ must be found so that these probabilities are equal to the specified significance level α.

The probability for the upper bound is given by

$$\alpha \;=\; P(X_{\mu_{\max}} \leq x) \;=\; P(Y_{\mu_{\max}} \leq \ln(x)) \;=\; P\left(Z \leq \frac{\ln(x) - \mu_{\max}}{a + re^{-\lambda\mu_{\max}}}\right)$$

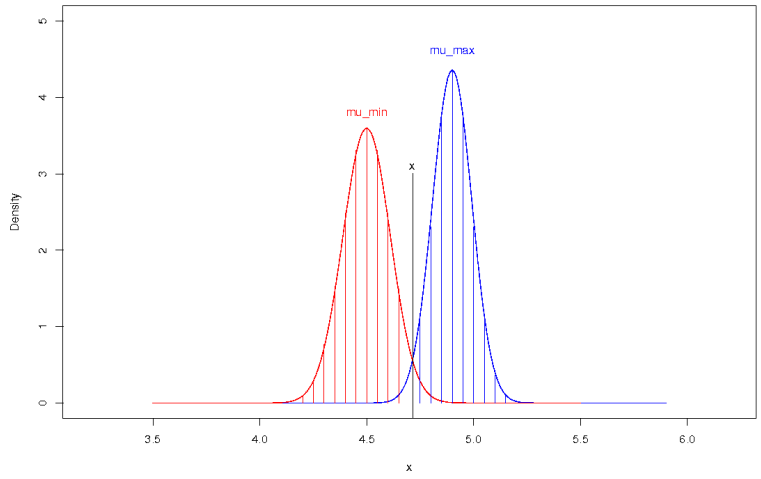


Figure 3.14: Determination of the lower and the upper bound of the intensity interval. $Y_{\mu_{\max}}$ and $Y_{\mu_{\min}}$ are able to produce the intensity $ln(x)$ with certain probabilities. $\mu_{\max}$ and $\mu_{\min}$ must be found so that these probabilities are equal to the specified significance level α.

where Z is the standard normal distribution with mean 0 and variance 1. Therefore, the equation

$$\Phi\left(\frac{\ln(x)-\mu_{\max}}{a+re^{-\lambda\mu_{\max}}}\right)-\alpha \;=\; 0 \tag{3.19}$$

has to be solved for $\mu_{\max}$. As in (3.17), Φ is the cumulative distribution function of the standard normal distribution. This is done again by simple bisection. One boundary is chosen as $\ln(x)$, the other one is determined by searching in the entourage of $\ln(x)$ for a change of the sign of (3.19).

Analogously, for the lower bound is required

$$\alpha \;=\; P(X_{\mu_{\min}} \geq x) \;=\; P(Y_{\mu_{\min}} \geq \ln(x)) \;=\; P\left(Z \geq \frac{\ln(x)-\mu_{\min}}{a+re^{-\lambda\mu_{\min}}}\right)$$

leading to

$$1-\Phi\left(\frac{\ln(x)-\mu_{\min}}{a+re^{-\lambda\mu_{\min}}}\right)-\alpha \;=\; 0 \tag{3.20}$$

(3.20) is solved by the same strategy as (3.19).

Note that α is not the probability that the true intensity lies in the interval $[e^{\mu_{\min}}, e^{\mu_{\max}}]$. It only means that a true intensity outside this range will produce a value like x with a probability lower than α.

3.3.6 Comparison with Other Approaches

Intensity dependent noise within iTRAQ^{TM} data was described in several studies in the last years. For example, Hu *et al.* (2006) recommended to discard low-intensity

spectra and identified the necessity for the incorporation of noise model analysis with comprehensive statistical robustness as described for microarray analysis.

Recently, Du *et al.* (2008) published an approach for modelling noise of mass spectrometry based proteomics. The noise in MS^1 is estimated by comparing expected and observed isotopic patterns based on well-known frequencies of isotopes of carbon which is assumed to follow a multinomial distribution. Additionally, for a Q-TOF device an intensity dependent part of the noise is described which is approximated by a Poisson distribution by the authors. The advantage of a Poisson based model is the absence of parameters (additionally to the intensity), hence no parameter estimation is necessary. Application of 95% interval based on the Poisson distribution to the test dataset shows that intensity dependent noise of the used Q-TOF device can also be modelled by Poisson (Figure 3.15).

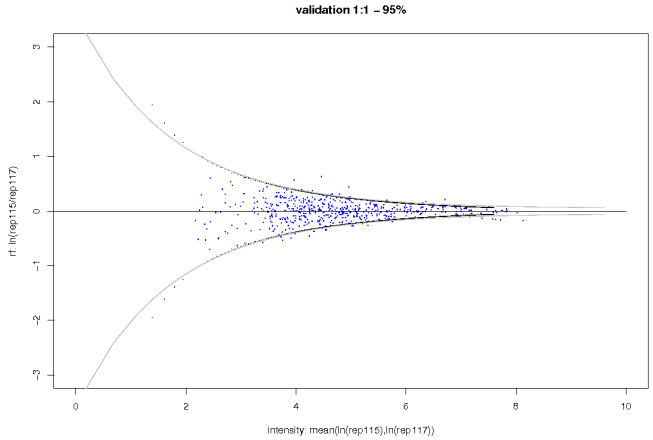


Figure 3.15: Calculation of a Poisson based 95% interval and application to the test dataset. Intensity dependent noise of iTRAQ^{TM} labelled samples analysed on a Q-TOF device alternatively can be approximated by a Poisson distribution.

However, using an Orbitrap XL device the observable noise characteristics are different and consequently, the 95% interval derived from the Poisson model does not fit the data as shown in Figure 3.16. The usage of a Poisson model for estimating the noise of the Q-TOF device may be beneficial due to lack of training, otherwise this model also requires the implementation of further parameters. Generally, the noise should be characterised using a scalable model comprising enough parameters in order to be applied to different workflows.

3.3.6.1 Alternative Models

In addition to the presented noise model $h(\mu; a, r, \lambda) = a + re^{-\lambda\mu}$ further models were investigated, which also might be able to simulate a representative distribution of iTRAQ^{TM} intensities. Investigation of 563 independent data points from

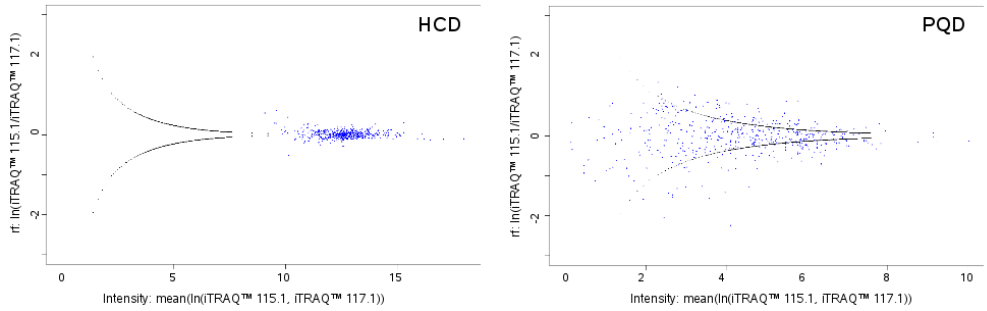


Figure 3.16: Calculation of Poisson based 95% interval and application to an unregulated sample analysed on Orbitrap XL using two different modes (HCD mode and PQD mode). Intensity dependent noise of iTRAQ^{TM} labelled samples analysed on an Orbitrap XL device can not be approximated by a Poisson distribution.

the test dataset calculates the best approximation regarding the expected 5% ratio of outliers (95% interval) for the presented model. Therefore, the established noise model is representative with a minor tendency to define some real regulations as non-regulated (2.84% instead of 5% outliers).

Table 3.7: Percentage of regulatory outliers (95% interval) calculated by different noise models and based on the test dataset.

$h(\mu;\theta)$	outliers [%]
$h(\mu;a,r,\lambda) = a + re^{-\lambda\mu}$	2.84
$h(\mu;a,r) = a + \frac{r}{\mu^2}$	30.73
$h(\mu;a,r) = a + \frac{r}{\mu+1}$	0.00
$h(\mu;a,r) = a + \frac{r}{\mu}$	0.00

3.3.6.2 Bayesian Statistics

Bayesian statistics traces back to Thomas Bayes (1702-1761). The centre of interest is built by Bayes' theorem allowing to estimate unknown parameters, to identify confidence intervals as well as hypothesis testing concerning these parameters. Traditional statistics considering unknown parameters as constants (instead of random variables) and determining probabilities by frequencies are not able to estimate constants not at random.

Bayesian statistics uses prior probabilities giving prior knowledge by a probability distribution. Bayes' theorem defines calculations with conditional probabilities

and is often used to compute posterior probabilities. Given events A and B Bayes' theorem is

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{3.21}$$

with

$P(A)$ = prior probability of A
$P(A|B)$ = conditional probability of A, given B ("posterior probability")
$P(B|A)$ = conditional probability of B given A
$P(B)$ = prior or marginal probability of B.

Bayes' theorem describes the way in which one's beliefs about observing A are updated by having observed B. For example, Bayes' theorem is applied in medicine when a test having a defined error probability returns a positive result ("diseased") and the probability of the existence of the disease is in demand.

Comparison with Bayesian Approach Baldi and Long (2001) developed a software tool (called Cyber-T) for the analysis of microarray expression data[2] based on Bayesian statistics. Cyber-T is a statistical program with a web interface that can be conveniently used on high-dimensional array data for the identification of statistically significant differentially expressed genes. It also contains a computational method for estimating experiment-wide false positive and negative levels based on the modelling of p-value distributions (PPDE).

In order to compare the established model with a Bayesian approach identical datasets are introduced to Cyber-T and the presented noise model. However, Cyber-T expects datasets containing two measurements of each unstimulated and stimulated samples. Proteomics is not able to comply with this condition since MS experiments never identify identical peptides and intensity heights are only relative that can not be compared to each other over several experiments. Therefore, a simulated dataset was built by generating pairs of unregulated as well as pairs of regulated noisy intensities according to the presented model. The simulated noisy intensities corresponding to every of 2000 true, unknown intensities (μ) are at a defined ratio of c: 80% of the tuples are unregulated ($c = 1$), each 5% are slightly regulated in a ratio (c) $1 : 1.25$, $1 : 1.5$, $1 : 2$ and $1 : 3$.

2000 4-tuples and 2000 2-tuples (1 x unregulated, 1 x regulated) were introduced to Cyber-T and the established noise model. Since the dataset based on known regulations, the identified regulations of both approaches are comparable. Based on the noise model's p-values (section 3.3.5) and Cyber-T's PPDEs the result is as listed in Table 3.8. The columns refer to the introduced ratio as well as the false

[2] http://cybert.microarray.ics.uci.edu/

positive and false negative identified results of Cyber-T and the new established noise model, respectively.

Table 3.8: Result of the comparison of the presented error probabilities of the established noise model and the Bayesian approach for the calculation of probabilities for differential expression (Cyber-T). The columns refer to the introduced ratio as well as the false positive and false negative identified results of Cyber-T and the new established noise model, respectively.

Ratio	Cyber-T	Noise Model $\sigma = a + re^{-\lambda\mu}$
1 : 1	0% false positive	0.75% false positive
1 : 1.25	73% false negative	62% false negative
1 : 1.5	36% false negative	27% false negative
1 : 2	9% false negative	10% false negative
1 : 3	2% false negative	2% false negative

The outcome clearly shows that the Bayesian approach in Cyber-T has no advantages concerning the validity of the existence of a regulation compared with the noise model presented in this work. The percentage of false positive and false negative results is higher for the Cyber-T software in comparison with the presented noise model.

4 Identification of Significant Regulations

In section 2.3 an intuitive method for the calculation of regulation factors at the level of peptides was introduced by dividing the measured iTRAQ^{TM} ion intensities. However, this idea does not consider the different quality of intensities presented in the previous section. Furthermore, a concept for the calculation of the most suitable regulation factor of any group of peptides is necessary. One possible application of this approach is the calculation of representative protein regulation factors based on individual peptide information. Weighted calculation of a regulation factor concerning an individual peptide considering all measurements of this peptide is required if it was detected several times comprising different regulation factors or if several MS/MS experiments are merged. Therefore, the calculation of the average regulation of several peptides and even the calculation of the regulation of single peptides must be performed in an intensity dependent manner.

Although Boehm *et al.* (2007) regard intensity dependent noise and take error estimations into consideration, they provide no strategy for intensity dependent weighting while calculating peptide and protein expression ratios. Actually, in the case of a large number of peptides, the median peptide expression ratio is recommended as protein expression ratio. Lin *et al.* (2006) deal with impreciseness of regulation factors derived from low intensity signals by simply defining a minimum intensity threshold. The remaining peptide expression ratios are determined by dividing iTRAQ^{TM} ion intensities whereas the protein expression ratios are derived from calculating a weighted sum of the peptide expression ratios. The peptides are weighted corresponding to the percentage of the sum of their iTRAQ^{TM} ion intensities regarding the sum of iTRAQ^{TM} ion intensities of all peptides belonging to one protein. Protein expression ratios calculated according to this approach are similar to those presented in the following.

As an application of the presented noise model an intuitive concept was developed for the visualisation-aided exploration of regulatory iTRAQ^{TM} data based on likelihood curves precisely depicting the overall data dependent quality of regulatory information (Hundertmark *et al.*, 2008). This approach can be applied to both, the peptide and the protein level: (i) one or more identical peptides resulting

in peptide likelihood curves and (ii) different peptides of a protein resulting in a protein likelihood curve as illustrated in figure 4.1.

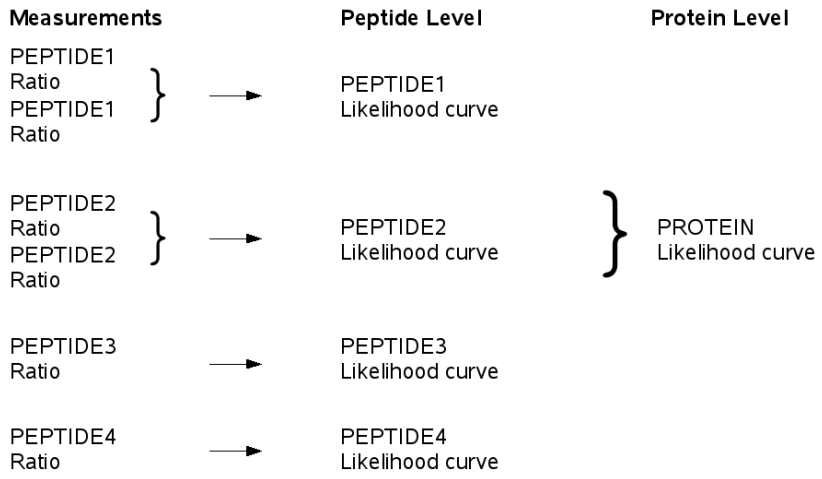


Figure 4.1: Levels of information: Regulation can be presented on the level of the measurement by calculation of ratios (division of measured intensities) and on the peptide level as well as on the protein level by calculation of the overall regulatory information.

4.1 Calculation of Regulatory Information

The expression ratio of a single peptide can be simply determined by dividing the iTRAQ^{TM} ion intensities if only one fragmentation experiment (MS/MS) was generated within an experiment. However, and as discussed in Hu *et al.* (2006), a different approach is required if multiple MS/MS spectra match to one peptide ("one peptide was identified several times"). According to the noise model, expression ratios derived from high intensities are more representative compared to those derived from low intensities. Therefore, reporter ion ratios that all have to be correlated to the same peptide should be integrated in a weighted manner depending on their reporter intensities and their corresponding signal qualities. In order to find the most likely expression ratio for a peptide, all possible expression ratios c_j between the lowest (c_{min}) and the highest ratio of iTRAQ^{TM} ion intensities (c_{max}) are discretised before they are related to all MS/MS spectra that match the actual considered peptide.

For each expression ratio c_j and each measured pair of intensities (x_i, y_i), $1 \leq i \leq n$, n = number of matched MS/MS spectra, a suitable pair of intensities is calculated by setting $\mu_{x_i} = c_j \mu_{y_i}$ where μ_{x_i} and μ_{y_i} are the true, unknown intensities of the measured intensities x_i and y_i, respectively. To find the best expression ratio c_j for all n matched MS/MS spectra the likelihood function

$$L(x, y, c_j) = \prod_{i=1}^{n} \left(\frac{1}{\sqrt{2\pi}\sigma_{x_i}} e^{-\left(\frac{x_i - c_j \mu_{y_i}}{2\sigma_{x_i}}\right)^2} \frac{1}{\sqrt{2\pi}\sigma_{y_i}} e^{-\left(\frac{y_i - \mu_{y_i}}{2\sigma_{y_i}}\right)^2} \right) \tag{4.1}$$

is computed with $\sigma_x = a + re^{-\lambda\mu_x}$ and $\sigma_y = a + re^{-\lambda\mu_y}$.

Logarithmic transformation and omitting the constant parts lead to the log-likelihood function

$$\tilde{L}(x, y, c_j) = \sum_{i=1}^{n} \left(-\ln(\sigma_{x_i}) - \left(\frac{x_i - c_j \mu_{y_i}}{2\sigma_{x_i}}\right)^2 - \ln(\sigma_{y_i}) - \left(\frac{y_i - \mu_{y_i}}{2\sigma_{y_i}}\right)^2 \right). \tag{4.2}$$

The best suitable overall expression ratio c_j for all matched MS/MS spectra of a multiple found peptide according to the noise model is the ratio resulting in the maximum log-likelihood according to (4.2).

The calculation of a protein expression ratio takes place in the same manner. Instead of regarding the expression ratios of a (multiple) matched peptide, the regulation factor which is best representing all peptides of the observed protein is computed. This is done by calculating the likelihoods for all regulation factors c_j to be considered ($c_{min} \leq c_j \leq c_{max}$ and a sufficiently small step size for c_j) and choosing the one resulting in the maximum likelihood. Hence, there are

n = total number of all MS/MS spectra matched to the regarded peptide(s)
c_{min} = lowest expression ratio of a matched MS/MS spectrum
c_{max} = highest expression ratio of a matched MS/MS spectrum

for the estimation of the expression ratio.

For visualisation purposes (section 4.2) it is important, to have symmetric ranges of possible regulation factors, i.e. regulations in $[0 \ldots 1]$ are transformed to $[-\infty \ldots -1]$ by replacing the measured intensities and adding a negative sign. For instance, if intensities 100 and 200 are measured corresponding to iTRAQ$^{\text{TM}}$ labels 115.1 and 117.1, the resulting regulation factor after dividing the intensity of label 115.1 by that of label 117.1 is $\frac{1}{2}$. Transformation is performed by dividing the intensity of label 117.1 by that of label 115.1 and adding a negative sign. Therefore, the outcome is $-\frac{200}{100} = -2$.

Regulation factors of single peptides are calculated this way as well. Consequently, if the intensities are extremely low, the resulting expression ratio is slightly deviating from the ratio of the intensities since usage of higher intensities results in a higher maximum likelihood due to decreasing noise at increasing intensity.

Thus, this approach can integrate information of all experimentally observed results without the necessity to reject data by arbitrary threshold levels. The quality of each iTRAQ$^{\text{TM}}$ reporter intensity can be specified by the established

noise model based on its individual intensity. Low intensity reporters in combination with those of better signal qualities, i.e. higher intensities, have only a minor effect on the final result, and vice versa.

An example of the resulting regulation factors calculated according to (4.2) is given below. 21 identified peptides and corresponding iTRAQ^{TM} ion intensities of the protein GSK3α from HGF stimulated cells in comparison with unstimulated cells are listed in Table 4.1. Column "Ratio" refers to the ratios obtained by dividing ion intensities iTRAQ^{TM} 115.1 and 117.1, column "MLE Ratio" refers to the regulation factor calculated according to (4.2). For comparison, the column "Mean" gives the mean value of the iTRAQ^{TM} ion intensity ratios in case the peptide was matched several times. The calculated protein ratio (MLE) of GSK3α in this example is -1.26 (not listed in Table 4.1).

Table 4.1: Peptides of GSK3α from HGF stimulated cells in comparison with unstimulated cells (Hundertmark *et al.*, 2008). Columns refer to peptide sequence, iTRAQ^{TM} 115.1 ion intensity, iTRAQ^{TM} 117.1 ion intensity, ratio obtained by dividing iTRAQ^{TM} 117.1 ion intensity and iTRAQ^{TM} 115.1 ion intensity, MLE regulation factor according to (4.2) and the mean value of iTRAQ^{TM} ion intensity ratios (only if a peptide was matched several times). Phosphorylated amino acids are coloured.

Peptide sequence	iTRAQ^{TM} 115.1	iTRAQ^{TM} 117.1	Ratio	MLE Ratio	Mean
YFFYSSGEK	215.87	212.66	-1.02	**-1.18**	-1.20
YFFYSSGEK	693.28	500.58	-1.38	**-1.18**	-1.20
GEPNVSYICSR	3969.56	3073.30	-1.29	**-1.29**	
SQEVAYTDIK	6415.65	3593.18	-1.79	**-1.78**	
GEPNVSYICSR	4393.51	3302.95	-1.33	**-1.33**	
GEPNVSYICSR	1770.04	1396.09	-1.27	**-1.27**	
VTTVVATLGQGPER	2677.79	2324.79	-1.15	**-1.15**	
GEPNVSYICSR	512.76	489.10	-1.05	**-1.05**	
VIGNGSFGVVYQAR	9501.10	7655.79	-1.24	**-1.24**	
SLAYIHSQGVCHR	1341.66	1128.88	-1.19	**-1.19**	
DIKPQNLLVDPDTAVLK	23.29	16.46	-1.41	**-1.15**	-1.46
DIKPQNLLVDPDTAVLK	466.03	381.36	-1.22	**-1.15**	-1.46
DIKPQNLLVDPDTAVLK	10.27	6.05	-1.70	**-1.15**	-1.46
DIKPQNLLVDPDTAVLK	460.73	419.40	-1.10	**-1.15**	-1.46
DIKPQNLLVDPDTAVLK	87.00	105.39	1.21	**-1.15**	-1.46
DIKPQNLLVDPDTAVLK	37.81	23.89	-1.58	**-1.15**	-1.46
DIKPQNLLVDPDTAVLK	56.18	34.64	-1.62	**-1.15**	-1.46
DIKPQNLLVDPDTAVLK	18.64	10.25	-1.82	**-1.15**	-1.46
TPPEAIALCSSLLEYTPSSR	4.70	2.17	-2.17	**-1.42**	
TSSFAEPG...GGGK	405.13	1289.34	3.18	**3.15**	3.16
TSSFAEPG...GGGK	220.62	690.21	3.13	**3.15**	3.16

As illustrated in Table 4.1, the peptide DIKPQNLLVDPDTAVLK was identified eight times revealing iTRAQ^{TM} reporter ion intensities between 2.17 and 466.03 and expression ratios (derived by division) between -1.82 and -1.10. The MLE calculated regulation factor on the one hand is -1.15, the mean value of the MLE

calculated peptide ratios on the other hand is −1.46. Calculation of MLE ratio is weighted in favour of high intensity measurements. Consequently, the impact of the last entry of `DIKPQNLLVDPDTAVLK` referring to iTRAQ^{TM} ion intensities 18.64 and 10.25 and a resulting expression ratio −1.82 is low. However, the impact of the measurements of the ion intensities 466.03 and 381.36 as well as 460.73 and 419.40 resulting in ratios −1.22 and −1.10, respectively, is high. Actually, the MLE ratio is calculated as −1.15 which is very close to −1.22 and −1.10.

The MLE ratio of the peptide `TPPEAIALCSSLLEYTPSSR` strongly deviates from the ratio determined by dividing the ion intensities (−1.42 instead of −2.17). The reason is mentioned before: due to increasing deviations in decreasing intensity ranges the very low signal intensities (4.70 and 2.17) result in lower likelihoods than slightly changed signal intensities.

4.2 Visualisation of Regulatory Information

All likelihood values l_j are plotted against the corresponding expression ratios c_j using the Java chart library JFreeChart[1]. Due to area normalising ($\int_{c_{min}}^{c_{max}} = 1$) the robustness of the underlying data is proportional to both the height and the slope of the produced curves which are called "likelihood curves" in the following. A likelihood curve represents the likelihoods (y-axis) of the most probable regulation factor – given by the maximum likelihood – as well as alternative regulation factors (x-axis). Therefore, the range of regulations is strongly limited within narrow curves in contrast to plain curves providing a wide range of regulations. In addition to the visual presentation, the information on robustness of the underlying data is given by the interval of robustness (IR) comprising the minimum interval length which contains 80% of the area under the corresponding curve. According to the range of possible regulation factors the range $[-1 \cdots +1]$ is omitted.

Figure 4.2 shows the peptide likelihood plot of the protein GSK3α. Each peptide is represented by a separate likelihood curve. The 80% IR is given in the right top side of the plot. Most of the regulatory information is very robust, reflected in IR values near 0.2. However, the reliability of the regulatory information on the peptide `TPPEAIALCSSLLEYTPSSR` (intensities 4.70 and 2.17) is of lowest quality since the probability of all regulation factors between −3.5 and 1 is nearly identical, resulting in an IR value of 4.25.

[1] http://www.jfree.org/jfreechart/

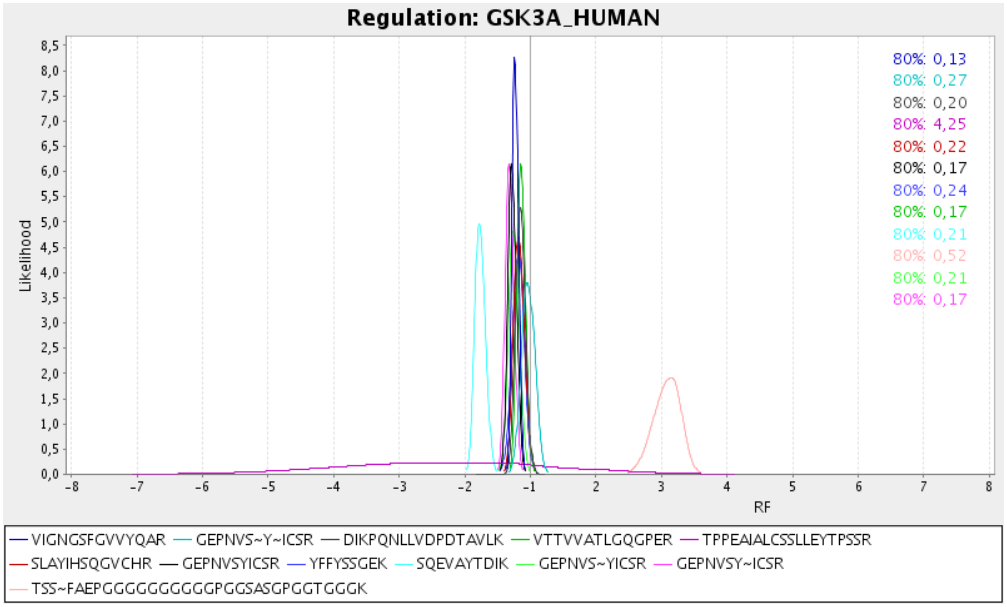


Figure 4.2: Peptide likelihood plot of the protein GSK3α from HGF stimulated cells. Each peptide is represented by a separate likelihood curve. Most of the peptides are slightly downregulated, one peptide is significantly downregulated and another one is significantly upregulated. The flat curve refers to a peptide of lowest signal intensities. Legends at the bottom and the right top side of the plot refer to the displayed peptide sequences and intervals of robustness (IR), respectively.

Depending on the aim of the analysis various views of a protein may be useful. All peptides of the protein can alternatively be represented separately by individual curves (peptide view) or be combined and visualised by a shared curve (protein view) within the plot. While Figure 4.2 shows the peptide view of GSK3α, Figure 4.3 illustrates the protein view of the same protein where a single likelihood curve represents the total protein. Finally, both kinds of information can be combined in such a way that individual peptides can be plotted separately in contrast to the remaining peptides of the protein (e.g. modified and unmodified peptides). Figure 4.4 illustrates this by presenting single curves corresponding to phosphopeptides and a shared protein curve (red) giving the unphosphorylated peptides. Omitting the modified peptides for the calculation of the protein curve results in a marginal shift of the curve towards a heavier downregulation.

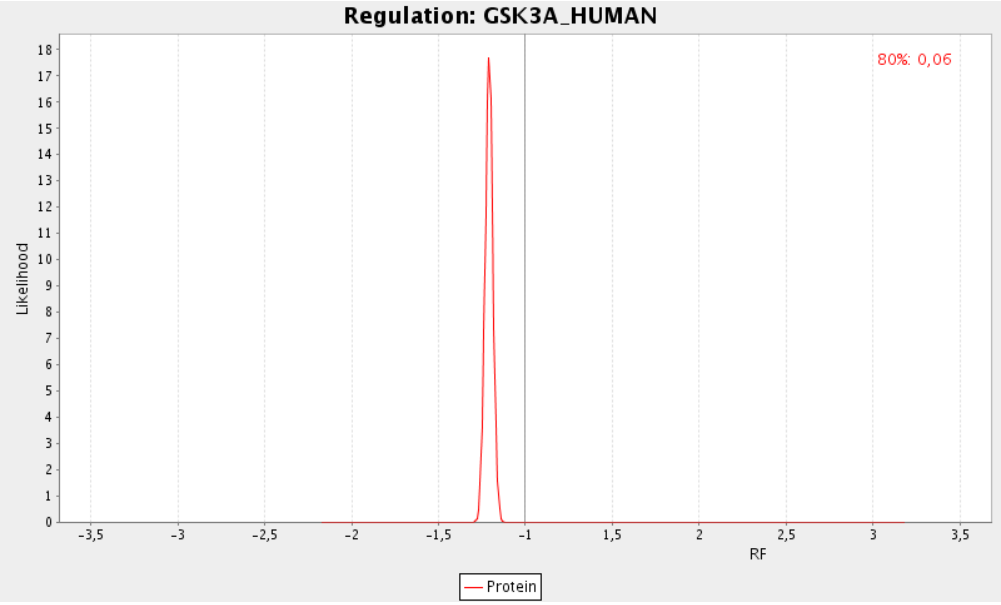


Figure 4.3: Protein likelihood plot of the protein GSK3α. All peptides are represented by a shared likelihood curve. Since most of the peptides are regulated very similar (compare Figure 4.2) the resulting protein curve is extremely narrow (80% IR = 0.06).

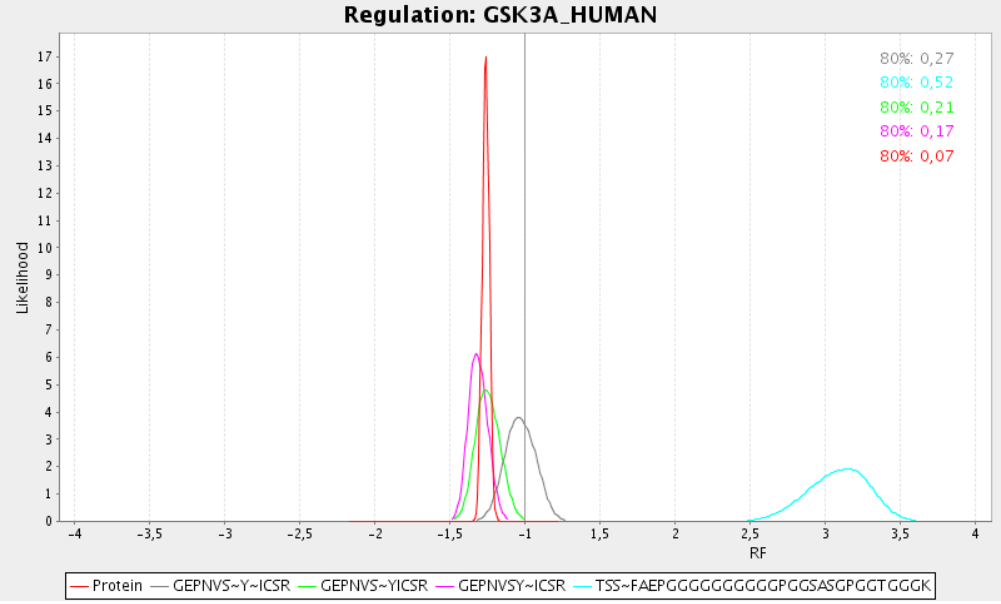


Figure 4.4: Mixed likelihood plot of the protein GSK3α. Each phosphopeptide is represented by a separate likelihood curve, non-phosphopeptides are represented by the red curve. Legends at the bottom and the right top side of the plot refer to the displayed peptide sequences and intervals of robustness (IR), respectively.

4.3 Development of the iTRAQassist Web Application

The strategies presented for the calculation and visualisation of regulatory information have been implemented as a web-based service for in-house analyses called "iTRAQassist". The program workflow includes preprocessing as described in section 3.1 performing improved peak detection, correction of isotopic impurities, sample normalisation and logarithmic transformation. Afterwards, the calculation of intensity intervals and error probabilities of contrary regulation (section 3.3.5) is performed before computing and visualising the regulatory information. Since large experiments result in very long program run times and need lots of memory for the calculations the application was developed as a web application in order to avoid blockages of desktop computers. The usage of Java for programming the calculations requires Apache Tomcat[2] (V5.5) and Java Servlet as well as Java Server Page technologies.

The iTRAQassist webinterface (Figure 4.5) allows to choose between different program settings, e.g. consideration of peptide phosphorylation, preferring peptide or protein view, definition of an enlarged mass range for peak detection as well as discarding low identified peptides (Mascot Peptide Score). Besides specifying the applied iTRAQ^{TM} reagents, essential information concerning the files containing the experimental results (Mascot .dat file) and the measurements of isotopic impurities (containing comma separated values) must be entered by the user.

[2]http://tomcat.apache.org/

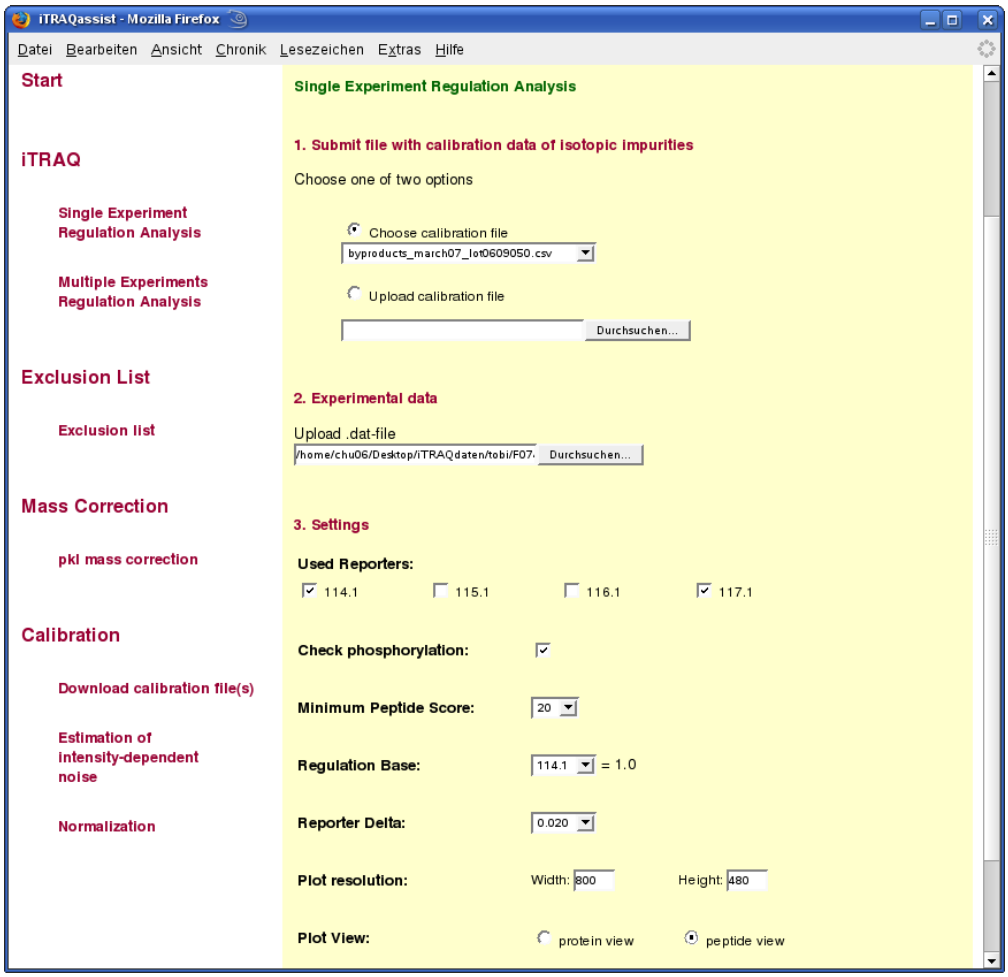


Figure 4.5: Detail of the iTRAQassist webinterface requiring several settings, e.g. uploading the experimental result file, selecting a result from the measurements of isotopic impurities and specification of applied iTRAQ^{TM} reagents.

Program run time strongly depends on the number of peptides detected in the MS experiments whereas the calculation of the likelihoods for possible regulation factors requires most of the time. Large result files comprising $>$ 1000 peptides often result in program run times $>$ 30 minutes (Intel Pentium D, 3.2 GHz (Dual Core), 2 GB RAM). The results consist of two parts: likelihood plots are available within the HTML result file as well as an additional archive. Furthermore, a result file can be downloaded (Excel format) containing all available information (peptide sequences, raw and corrected ion intensities, intensity interval, error probabilities, most suitable protein as well as peptide regulation factors, normalisation factor, etc). Figure 4.6 shows an extract of the result of running iTRAQassist.

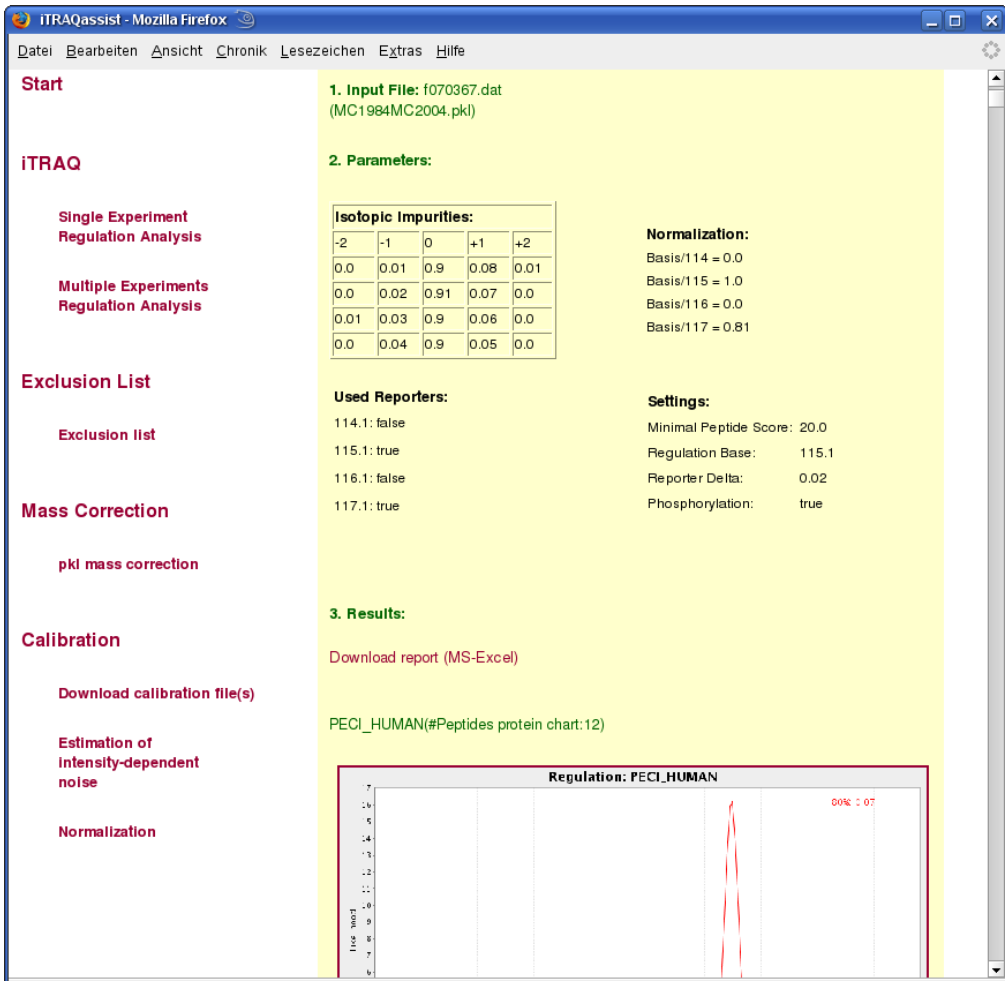


Figure 4.6: Result of running iTRAQassist: Information on the experimental file, isotopic impurities, normalisation factor, the applied iTRAQ^{TM} reagents and several settings defined by the user followed by the link for result file generation (Excel format), and the presentation of the likelihood plots.

A variation of the iTRAQassist software was developed for the calculation of regulation factors across several experiments (May (2007)). Preprocessed iTRAQ^{TM} ion intensities of identic peptides from iTRAQassist result files (Excel format) of different experiments are introduced in the calculation of the most suitable and alternative regulation factors according to (4.2). Again, the result of the regulation analysis of multiple experiments are likelihood plots and a result file (Excel format) similar to the result file of the single experiment analysis presented before. Figure 4.7 shows an extract of the webinterface for analysing the regulatory information of several experiments by the iTRAQassist software. The iTRAQassist result files are introduced containing all necessary information unless the mapping of the used iTRAQ^{TM} labels and the experimental conditions.

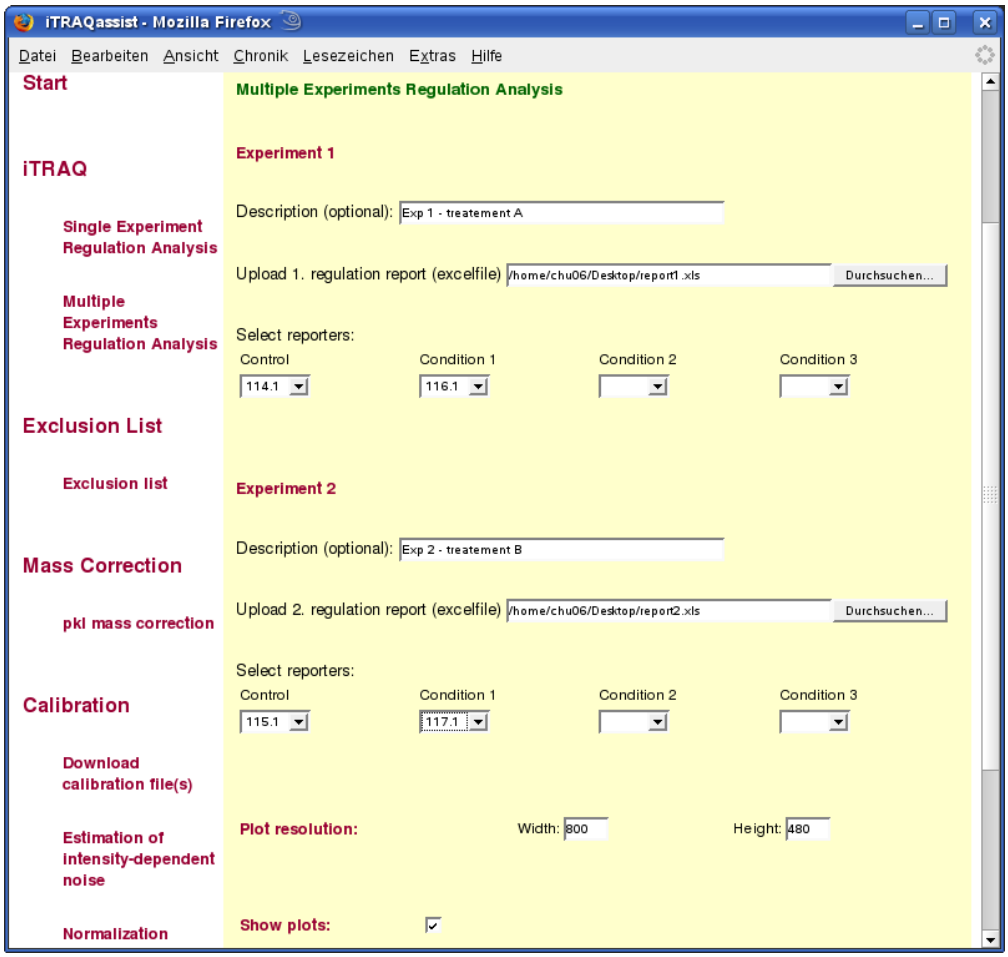


Figure 4.7: Detail of the webinterface for analysing the regulatory information of multiple experiments by iTRAQassist software. Besides iTRAQassist result files from single experiment analysis the mapping of used iTRAQ^{TM} labels and the experimental conditions are required.

5 Detection of Post-translational Modifications

Figure 4.4 from section 4.2 demonstrates the ability of likelihood plots to relate regulatory information on specific peptides on the one hand and the remaining peptides of the same protein on the other hand. Regulatory information on the remaining peptides is represented either by a combined curve or by individual peptide curves. Both approaches are basically suitable for comparing the regulatory information on specific peptides and the remaining peptides within the same protein. Mostly, varieties concerning the regulatory information on all peptides belonging to one protein is low, however, in the case of post-translational modified peptides regulatory information may differ significantly. Besides the possibility of false-positive peptide to protein assignments and errors in measurements the existence of modifications is the most frequent cause for wide differences concerning the regulatory information on peptides within the same protein.

5.1 Detection of Post-translational Modifications by Mass Spectrometry

Peptide sequencing by mass spectrometry is based on mass differences of the detected ions (for details see section 2.2.2). Modifications that change the mass of an amino acid (for example by adding functional groups e.g. phosphate groups) can result in failure of peptide sequencing if the corresponding mass delta is not considered as a variable modification. Because of the high variety (> 200) of known post-translational modifications consideration of all potential PTM in parallel is not possible due to combinatorial and computational complexity. Therefore, only those modifications are identified by mass spectrometry that the analysis is optimised for.

5.2 Peptide Likelihood Curves for the Identification of PTM

As mentioned before, modified peptides may be regulated significantly within a differently regulated protein. Those peptides are named "outliers" in the following. In several studies, unmodified peptides corresponding to oppositely regulated modified peptides can be detected. This observation is not surprising since the proportion of an unmodified peptide which is equivalently present in two samples decreases if a significant amount of the peptide is modified. Figure 5.1 illustrates this behaviour. Sample 1 and sample 2 contain equal amounts of the peptide SSTVTEAPIAVVTSR. After stimulation (sample 2) half of the unmodified present peptide is phosphorylated. The resulting regulation factors (RF) are $\frac{4}{1} = 4$ in the case of the phosphopeptide and $\frac{3}{6} = \frac{1}{2} \hat{=} -2$ in the case of the unmodified peptide.

Sample 1	**Sample 2**
SS∼TVTEAPIAVVTSR	SS∼TVTEAPIAVVTSR
SSTVTEAPIAVVTSR	SS∼TVTEAPIAVVTSR
SSTVTEAPIAVVTSR	SS∼TVTEAPIAVVTSR
SSTVTEAPIAVVTSR	SS∼TVTEAPIAVVTSR
SSTVTEAPIAVVTSR	SSTVTEAPIAVVTSR
SSTVTEAPIAVVTSR	SSTVTEAPIAVVTSR
SSTVTEAPIAVVTSR	SSTVTEAPIAVVTSR

Figure 5.1: Opposite regulation of the corresponding unmodified peptide if it is modified resulting in regulation in one of two samples. The resulting regulation factor of the modified peptide is 4 and that of the unmodified peptide is -2.

Figure 5.2 illustrates a very clear example of a protein containing both upregulated phosphopeptides (modified at two neigbouring amino acids) and the corresponding downregulated unmodified peptide. The upregulated phosphopeptides are represented by the green (RF = 4.66), purple (RF = 4.52) and grey (RF = 19.03) curves, the corresponding unmodified peptide (turquoise) is downregulated (RF = -5.96).

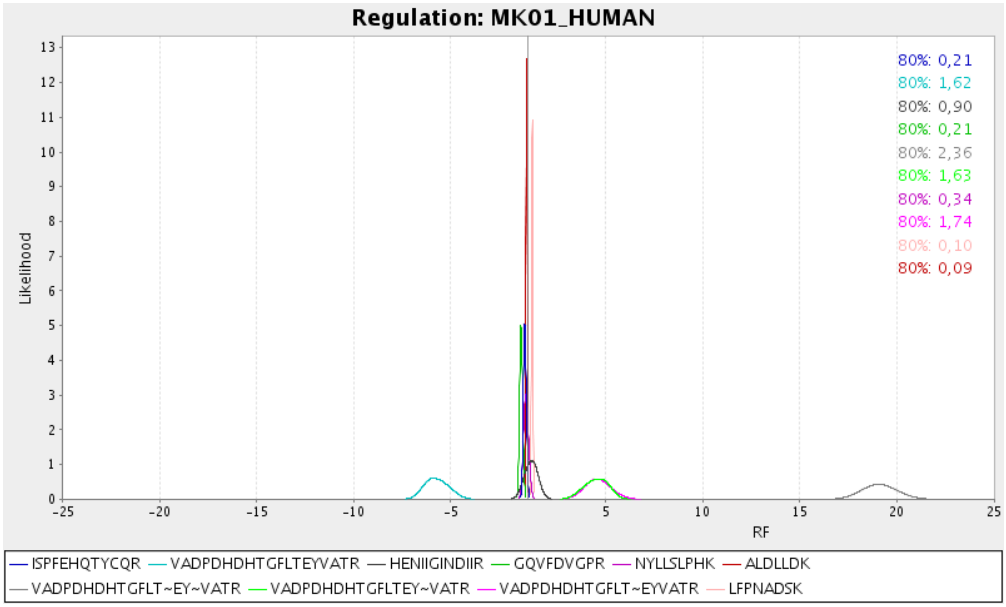


Figure 5.2: Likelihood plot of the protein MK01 containing one differently phosphorylated upregulated peptide (green, purple, grey) and the corresponding downregulated unmodified peptide (turquoise).

The example above shows the opposite expression of modified and unmodified variants of one peptide within the same protein. Depending on the existing quantities of modified and unmodified peptides, this relation can be more or less pronounced – in many cases the expression of the unmodified peptide is not changed at all due to low abundance of modified peptide. However, if the corresponding unmodified peptide is regulated diffferently than the remaining unmodified peptides of a protein, then this could be a starting point for the investigation of post-translational modifications.

A promising new approach for the detection of such outlying peptides concerning their regulatory information is cluster analysis of peptide likelihood curves belonging to the same protein. In the case of discovering a single peptide which is significantly regulated differentially within a protein and fulfilling some further conditions (for details see section 5.2.1), this peptide could be the target of a known or unknown modification. Unfortunately, the existence of a modification is not the only possible cause for the observed behaviour, expression regulation of protein isoforms could induce a similar effect. A protein isoform is a version of a protein with only small differences in sequence and function to another version of the same protein. Different forms of a protein may be produced from different but related genes, or may arise from the same gene by alternative splicing which is a method to exchange the sequence of gene products. Partial protein degradation by cellular enzymes (so called “proteolysis”) which is a PTM as well and protein

isoforms may yield similar results: In both cases, the regarded peptide may be detected more frequently – even exclusively – in one of the samples.

From these considerations a strategy for the detection of outliers can be derived that returns a list of potential modified peptides.

5.2.1 Strategy for the Detection of PTM

First of all, analysis of the regulatory information is performed according to the established noise model resulting in peptide likelihood curves. In order to allow the identification of outliers a minimum number of four peptides must be assigned to every further considered protein. The identification of outlying peptides is not possible if the minimum number of peptides is lacking that the outlier is related to. Proteins have to be analysed by a clustering approach which returns clusters of peptides concerning their regulatory information. Every single peptide forming its own cluster – in the following called "single cluster peptide" – is considered to be a regulatory outlier and consequently, might represent a PTM regulated protein region.

If this strategy suggests one peptide as an outlier, using EBI's webservice client WSDbFetch[1] known modifications are queried from UniProtKB database[2] corresponding to each remaining single cluster peptide. This draft of a workflow forms a first basis for screening large datasets for unknown modified peptides. Upon introducing general clustering approaches and comparison of several specific clustering algorithms adapted to the existing data, first results are given in section 5.3.3.3. Analysis of protein isoforms and peptide to protein assignments are not included.

5.3 Cluster Analysis

Cluster analysis (clustering) is the process of grouping data into classes or clusters so that objects within a cluster have high similarity to each other in comparison to objects that belong to different clusters. By the means of cluster analysis patterns within datasets are to be found. Cluster analysis is an unsupervised learning technique. This means that the classification of the objects is performed without knowledge about the available classes. Similarities and dissimilarities of the object are based on attribute values describing the data. Often, distance measures are used for the characterisation of (dis)similarities.

[1] http://www.ebi.ac.uk/Tools/webservices/downloads/java/lib.zip, 11.08.2008
[2] http://expasy.org/sprot/, 11.08.2008

5.3.1 Introduction into Cluster Analysis

Most of the clustering techniques are either hierarchical or partitioning clustering algorithms. Hierarchical algorithms create a hierarchical decomposition of the given set of data objects. They can be agglomerative ("bottom-up") or divisive ("top-down"). Agglomerative algorithms start with clusters each containing one element and continue merging the clusters. Divisive algorithms start with one large cluster containing all elements in the dataset and continue splitting clusters. Hierarchical clustering generates data in a tree structure which is illustrated by dendrograms in general. Partitioning algorithms on the other hand construct a partition consisting of k subsets of the data where each subset represents a cluster. The most known representatives of partitioning clustering algorithms are k-means and its derivatives. k-means starts with randomly generating k clusters. In the next step, the cluster centres are determined and afterwards each point is assigned to the nearest cluster centre. Subsequently, the new cluster centres are recomputed. The previous steps are repeated until some convergence criterion is met. Further details can be found in A.L. Symeonidis (2005), J. Han (2001) and Guojun *et al.* (2007).

Within partitioning clustering, a distinction is drawn between deterministic clustering ("crisp clustering" or "hard clustering") and probabilistic clustering ("fuzzy clustering" or "soft clustering"). In contrast to deterministic clustering which assigns every object to exactly one cluster (membership degree u of object i and cluster j; $u_{ij} \in \{0, 1\}$) fuzzy clustering enables objects to belong to one or more clusters with membership degrees. In the case of membership = 1 and membership = 0 the object is assigned to the cluster either completely or not at all. The most common fuzzy clustering algorithm is fuzzy c-means which combines k-means and the fuzzy principle providing membership degrees between 0 and 1 (membership degree u of object j and cluster i; $u_{ij} \in [0, 1]$).

By means of a distance (similarity) measure the distance (similarity) of two elements can be determined. Since clusters are represented by objects, which are of the same structure as the elements to be clustered, distance and similarity measures are also used for comparisons of elements and clusters as well as comparisons between clusters. Anytime a cluster is updated the distances between the updated cluster and the remaining elements and clusters are to be refreshed as well. Some of the most popular methods for the calculation of the distances between two clusters $\mathcal{A}$ and $\mathcal{B}$ are "single linkage"

$$\min\{\, d(a, b) : a \in \mathcal{A},\, b \in \mathcal{B} \,\} \tag{5.1}$$

and "complete linkage"

$$\max\{\, d(a,b) : a \in \mathcal{A},\, b \in \mathcal{B} \,\} \tag{5.2}$$

with $d(a,b)$ = distance between elements a and b.

5.3.2 Fuzzy Clustering

Fuzzy clustering (probabilistic clustering) divides a dataset into a set of clusters and – in contrast to hard or deterministic clustering – a data object can be assigned to more than one cluster. In order to handle noisy and ambiguous data, membership degrees of the data to the clusters are computed. Most fuzzy clustering techniques are designed to optimise an objective function with constraints. The most common approach is the so-called probabilistic clustering with the objective function

$$f \;=\; \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^{m} d_{ij} \tag{5.3}$$

under the constraints

$$\sum_{i=1}^{c} u_{ij} \;=\; 1 \qquad \text{for all } j = 1, \ldots, n \text{ (n = number of objects).} \tag{5.4}$$

In this equation it is assumed that the number of clusters c is fixed. How to determine the number of clusters will be discussed later on in section 5.3.2.3. u_{ij} is the membership degree of the data object j to the ith cluster which depends on the distance of the object j and the cluster i. d_{ij} is a certain distance measure specifying the distance between data object j and cluster i, for instance the (squared) Euclidean distance of j to the ith cluster centre if the data objects are simple points, not likelihood curves as in the case of the investigations here. The parameter $m > 1$, called fuzzifier, controls how much clusters may overlap. The constraints (5.4) lead to the name probabilistic clustering, since in this case the membership degree u_{ij} can also be interpreted as the probability that j belongs to cluster i. The parameters to be optimised are the membership degrees u_{ij} and the cluster parameters which are hidden in the distances d_{ij}. Since this is a non-linear optimisation problem, the most common approach to minimise the objective function (5.3) is to alternatingly optimise either the membership degrees or the cluster parameters while considering the other parameter set as fixed. Assuming the cluster parameters and therefore the values d_{ij} as fixed, the best choice for the membership degrees is given by

$$u_{ij} \;=\; \frac{1}{\sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{kj}} \right)^{\frac{1}{m-1}}}. \tag{5.5}$$

If the object j is identical to one or more clusters ($d_{ij} = 0$ for one or more clusters), one must deviate from (5.5) and assign the object j with membership degree 1 to one of these clusters and set $u_{ij} = 0$ for the other clusters i.

The update equation for the cluster parameters or prototypes representing a cluster strongly depends on the type of the cluster. For the specific case of likelihood curves an algorithm is proposed in section 5.3.2.1.

Cluster validity measures are used to validate a clustering result in general and also to determine the number of clusters. In order to fulfil the latter task, the clustering might be carried out with different numbers of clusters and that one yielding the best value of the validity measure (which depends on the selected measure) is assumed to have the correct number of clusters.

A straight forward validity measure is the objective function (5.3) itself. However, (5.3) will always decrease with increasing number of clusters. Therefore, if the number of clusters is determined based on (5.3), the procedure is as follows. The number of clusters c is increased step by step starting from $c = 1$ and (5.3) is evaluated each time. As long as increasing the number of clusters leads to a significant decrease of (5.3), the optimum number of clusters is still not reached. Once (5.3) starts to drop slowly when c is increased, c is too high. A challenging task is to determine the last significant decrease which gives the correct number of clusters.

There are other validity measures like the partition coefficient and the partition entropy (Bezdek, 1981). Both these measures validate the clustering result based on the membership degrees only without taking specific properties of the cluster prototypes into account.

The partition coefficient is defined by

$$\frac{\sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^2}{n}. \tag{5.6}$$

The higher the value of the partition coefficient, the better the clustering result. The highest value 1 is obtained, when the fuzzy partition is actually crisp, i.e. $u_{ij} \in \{0, 1\}$. The lowest value $1/c$ is reached, when all data are assigned to all clusters with the same membership degree $1/c$. This means that a fuzzy clustering result is considered to be better, when it is more crisp.

The partition entropy is given by

$$-\frac{\sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}\ln(u_{ij})}{n} \tag{5.7}$$

The lower the value of the partition entropy, the better the clustering result. This means that similar to the partition coefficient crisper fuzzy partitions are considered to be better.

As mentioned before, there are many other validity measures for fuzzy clustering. However, they are not of interest here, since they assume the data to be points in $\mathbb{R}^p$ and not likelihood curves as in this case.

5.3.2.1 Prototype Based Fuzzy Clustering of Likelihood Curves

Clustering of curves requires the definition of a distance measure which is considerably more complex than comparing the positions of points in a coordinate system. Various kinds of distance measures for clustering likelihood curves can be imagined e.g. (i) the maximum peak position, (ii) the profile (width and height) of the curve or (iii) the overlapping area below compared curves. The selected distance measure serves for comparisons between each curve object and a cluster which is represented by a protoype curve. The prototype curve has the same properties as peptide likelihood curves (e.g. total area under the curve = 1) and consists of those area parts that are shared by the majority of assigned curves (for details see paragraph "Generation of Prototypes").

Since likelihood curves give the distribution for the true (unknown) position of the regulation factor, it would not be advisable to take this value exclusively as distance measure. Choosing the non-overlapping area of curves as distance measure combines the position of the regulation factor on the one hand and the profile of the curve on the other hand. Furthermore, possible uncertainty concerning the exact position of the regulation factor is taken into account.

It is assumed that the horizontal axis is divided into T intervals of equal length. Thus, the objective function is (5.3) and d_{ij} is given by

$$1 - \int_{-\infty}^{+\infty} \min\{y_j(t), v_i(t)\}dt \tag{5.8}$$

with y_j = peptide j, v_i = prototype i.

Discretisation leads to

$$d_{ij} = 1 - \sum_{k=1}^{T} \min\{y_j(t_k), v_i(t_k)\} \tag{5.9}$$

where $t_0, \ldots, t_T$ are the discretised regulation factors.

Areas under curves are normalised to 1, in consequence $0 \leq d_{ij} \leq 1$ and especially

$$d_{ij} = \begin{cases} 0 & \text{if peptide } i \text{ and prototype } j \text{ do overlap completely} \\ 1 & \text{if peptide } i \text{ and prototype } j \text{ do not overlap.} \end{cases}$$

Minimisation of the objective function f given by (5.3) is done by generation of new prototype curves from the former prototype curves and the total amount of peptide curves (for details see section 5.3.2.1). This process terminates when updating the prototypes yields no further decrease of f.

Generation of Prototypes For the identification of the best partition of all peptide curves into c clusters (c fixed) c prototypes are initialised firstly. Subsequently, for all prototypes $i \in [1 \dots c]$ and all peptides $j \in [1 \dots n]$ the membership degrees u_{ij} are calculated by (5.5). The initialisation and update scheme for the cluster prototypes is described in detail in the following.

The listing below gives a general idea of the algorithm.

```
result[];
for ( 1 ≤ c ≤ n ){
    f^(p_old) = ∞;
    p := initialise prototypes (curves, c);
    d := calculate distances (curves, p);
    u := calculate u (d);
    f^(p) := evaluate cluster (u, d);
    while ( |f^(p) − f^(p_old)| > ϵ ){
        p_old := p;
        f^(p_old) := f^(p);
        p := update prototypes (p, curves, u);
        d := calculate distances (curves, p);
        u := calculate u (d);
        f^(p) := evaluate cluster (u, d);
    }
    result[c-1] = p_old;
}
return result;
```

Initialisation In order to avoid unsuitable results based on an unfavourable initialisation step, repeated initialisation (number of repetitions $k = 3$) is preferred. Initialisation of c prototypes is done by randomly choosing c likelihood curves from the dataset which are slightly modified by multiplying the centre with a fixed factor and cutting off the edges when the size of the area $A = 1$ is reached. To obtain results as different as possible it is important that different combinations of peptide curves are selected in all of the k initialisation steps.

Updating The aim of repeated updating is the generation of a new set of prototypes from the previous set of prototypes. In the case of crisp clustering which means that the likelihood curve l_j either belongs to the cluster i ($u_{ij} = 1$) or not ($u_{ij} = 0$), the update procedure would be very simple: the more curves are overlapping at a position t, the more the objective function for the clustering will be reduced when the prototype curve has a high value at t as well. At first, areas with a high number of overlapping curves are added to the prototype. Step by step, less overlapping areas are added as well and the procedure is finished when the area below the prototype likelihood curve reaches the value 1. Therefore, the new prototype likelihood curve is composed of those areas where the majority of likelihood curves do overlap.

However, fuzzy clustering allows a likelihood curve, that does not match one of the available clusters perfectly, to be assigned to several clusters. Consequently, this curve influences marginally several clusters instead of deforming one single cluster which is a criterion for using fuzzy clustering. Furthermore, fuzzy clustering provides a number of validity measures for the identification of the number of clusters inherent in the data. Due to these benefits, fuzzy clustering is applied. Besides the number of overlapping curves the weight w_i

$$w_i^{(t)} = \sum_{j=1}^{n} u_{ij}^{m\ (t)} \tag{5.10}$$

strongly affects the development of a prototype i from likelihood curves l_j and the former prototype.

The update procedure for the actual prototype i is as follows: Initially, all points $p_j^{(t)}$ that are part of the likelihood curve j, $0 \leq j \leq n$, are weighted by application of equation (5.10) related to the considered prototype i. Therefore, points belonging to a likelihood curve which is similar to prototype i (high membership degree u_{ij}), get better values than those belonging to a curve that is less overlapping with prototype i. Furthermore, the weight is increasing with every additional curve which overlaps with curve j in the considered interval.

A simple heuristic strategy to add the most interesting points to the new prototype is the following: All of the points are sorted in decreasing order with respect to their weights, regardless of their belonging to a special peptide curve. Step by step, the points with the highest weights are added to the prototype likelihood curve. In order to avoid that two clusters are represented by the same prototype comprising two centres, every newly added point must be directly adjacent to the present dataset. This means that the x-coordinate of the new point x_p must not exceed the borders of the interval of the partially constructed likelihood curve $[x_{min}, ..., x_{max}]$ for more than one interval length l ($x_{min} - l \leq x_p \leq x_{max} + l$).

Figure 5.3 presents the resulting prototype after initialisation with the labelled curve and all possible updates. In this example, it is assumed that all six peptides build a single cluster ($c = 1$) and the prototype representing this cluster was calculated.

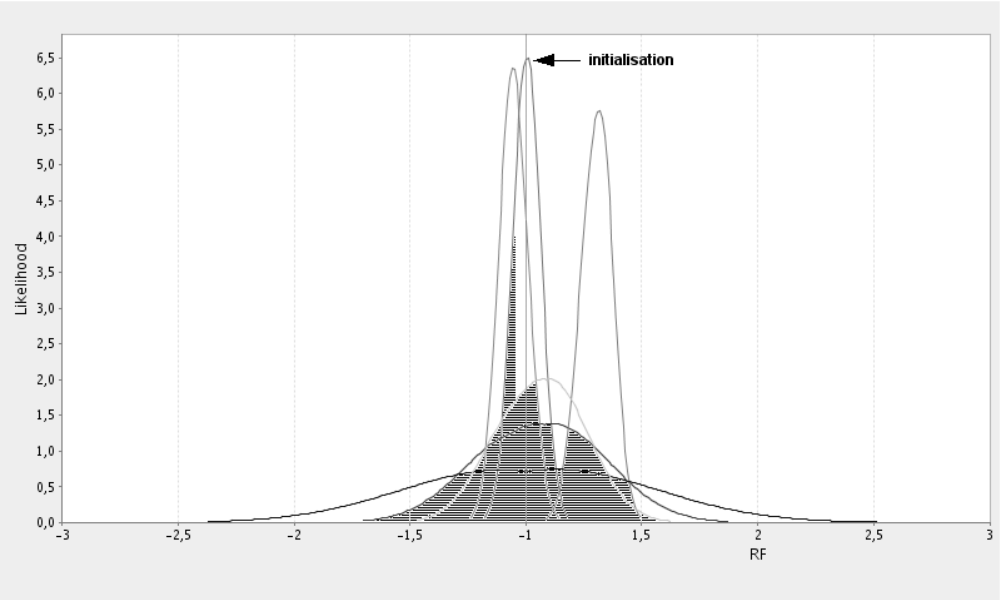


Figure 5.3: Protein PCTK1: Resulting prototype after initialisation with the labelled curve and all possible updates. The prototype likelihood curve mainly consists of those areas where most of the data likelihood curves overlap.

5.3.2.2 Results of Prototype Based Fuzzy Clustering

For identifying the optimal number of clusters the available validity measures "partition entropy", "partition coefficient" and "objective function" are analysed and the best assignment of peptide curves has to be found for this number of clusters.

Analysis of Validity Measures

In order to find out how the validity measures could be used for the identification of the right number of clusters, a plot of the function corresponding to every validity measure is generated. The results of the functions are plotted over the number of clusters.

Partition Coefficient The partition coefficient (for details see section 5.3.2 and (5.6)) should be maximsed at the optimum number of clusters c_{opt}. In the case of $c = 1$, the partition coefficient is always 1 since every element is assigned to the only existing cluster with membership degree $u_{ij} = 1$ (compare (5.7)). In the case of $c = n$ the partition coefficient has a very high value as well. Between these local

maxima, at least one minimum and a varying number of local maxima may occur. In general, the maximum value of c, $1 < c < n$, is of interest.

The peptide likelihood curves of a protein without differentially regulated peptides – which is the most common case – consist of only one cluster. Consequently, a maximum at $c > 1$ does not represent the optimum number of clusters. Therefore, the optimum number of clusters has to be determined by another mean which is the detection of the most significant increase after leaving a minimum. If all peptide curves are assigned to one cluster, the partition coefficient drops down from $c = 1$ to $c = 2$ and increases from then on.

Partition Entropy The behaviour of the partition entropy is directly opposed to the partition coefficient. According to (5.7) it always starts with 0 in the case of $c_{opt} = 1$ due to membership degrees $u_{ij} = 1$ and multiplying by $\ln u_{ij}$. With increasing number of clusters, the partition entropy is decreasing. The best number of clusters is given by the beginning of the decrease after leaving the maximum.

Objective Function The objective function is minimised. An increasing number of clusters results in a deacrease of the objective function. In order to illustrate this it is assumed, that n likelihood curves are assigned to c clusters. Addition of a further cluster causes that (at least) one curve is assigned to the new cluster yielding an improved (lower) distance value. The multiplicative impact of the decreasing distance value results in a deacrease of the objective function. By application of the so-called "elbow criterion" the corresponding function plot is analysed. The elbow criterion states that the optimal number of clusters is found as soon as adding another cluster adds no further significant information. Unfortunately, the increase from $c = 1$ to $c = 2$ is very strong in all cases. Hence, the validation of only one cluster is not possible.

Examples

In the following, two examples are presented in order to illustrate the analysis of validity measures and the results of fuzzy clustering. In each case, the topmost figure depicts the likelihood plot, in the middle are shown the function plots referring to the partition coefficient (left hand side) and to the partition entropy (right hand side). The plot of the objective function is placed at the bottom.

The first example shows the protein NEK9. In the peptide plot, 17 peptide likelihood curves are presented which cluster into more than four clusters. The analysis of the partition coefficient's plot (Figure 5.4 middle, left hand side) results proposing five or seven clusters, since the increases after the corresponding local minima are strongest. The plot of the partition entropy (Figure 5.4 middle, right

hand side) suggests to choose seven clusters, since the decrease after the corresponding local maximum is strongest, while the objective function (bottom) yields $c_{opt} = 5$. The assignment of the peptides by prototype based fuzzy clustering to five and seven clusters is presented in Table 5.1.

In conclusion, due to different cluster numbers this result is not satisfying. For example, by visual inspection the phosphopeptides `SSTVT`∼`EAPIAVVTSR` and `SST`∼`VTEAPIAVVTSR` are obviously well separated and each must build single-peptide clusters. The presented prototype based fuzzy clustering approach assigned both peptides into one common cluster in both possible results ($c_{opt} = 5$ and $c_{opt} = 7$).

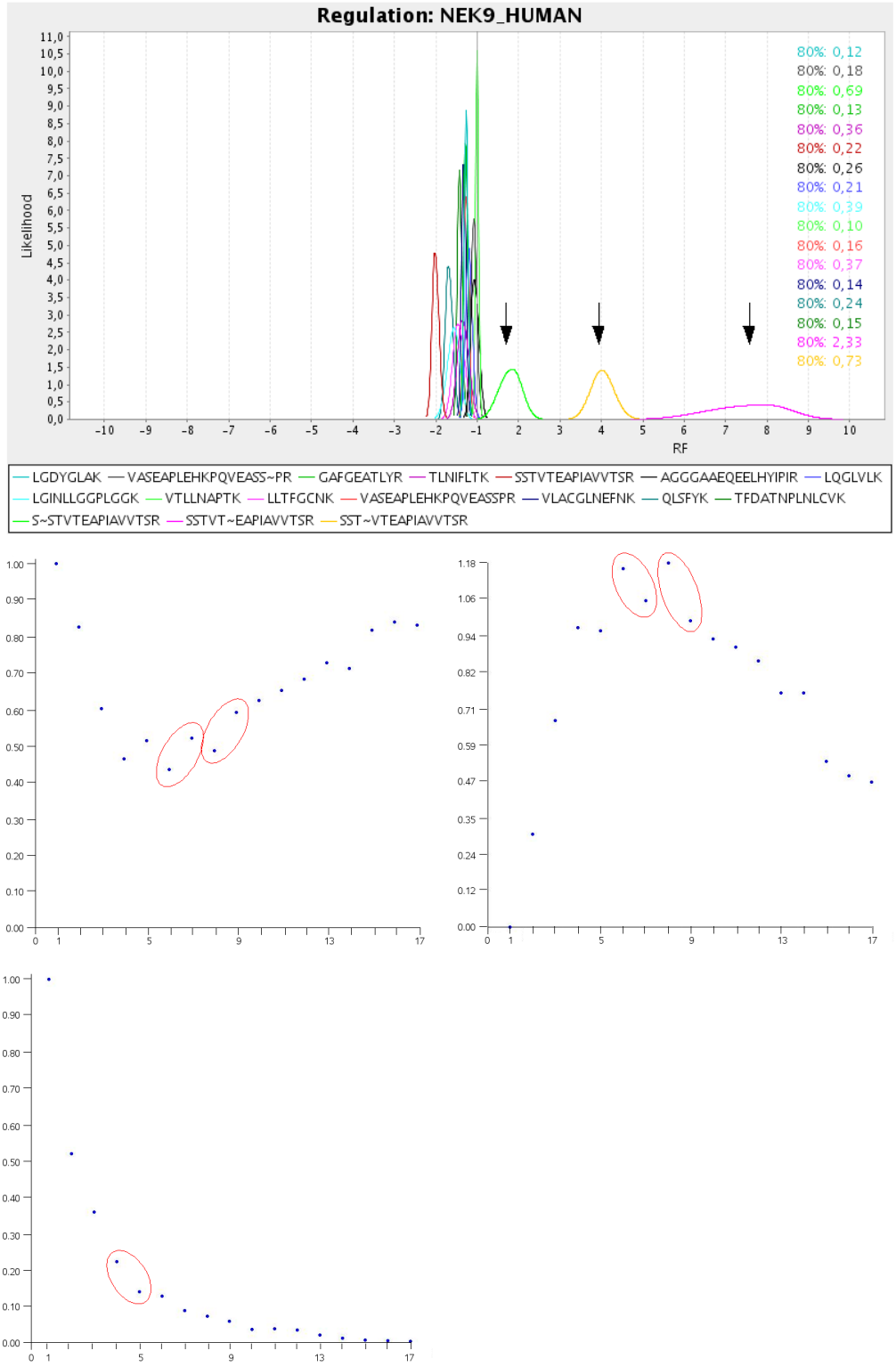


Figure 5.4: Fuzzy clustering results: Likelihood plot and validity measures of the protein NEK9. Top: Likelihood plot presenting 17 peptides clustering into ≥ 5 clusters. Middle: Function plot of the partition coefficient (left) and the partition entropy (right) indicating the existence of 5 or 7 clusters. Bottom: Plot of the objective function indicating the existence of 5 clusters.

Table 5.1: Assignment of peptides from Figure 5.4 to 5 and 7 clusters resulting from prototype based fuzzy clustering.

5 Cluster	**7 Cluster**
SSTVTEAPIAVVTSR	SSTVTEAPIAVVTSR
SSTVT∼EAPIAVVTSR, SST∼VTEAPIAVVTSR	SSTVT∼EAPIAVVTSR, SST∼VTEAPIAVVTSR
LGINLLGGPLGGK, LLTFGCNK, QLSFYK, TFDATNPLNLCVK, TLNIFLTK	LQGLVLK, VASEAPLEHKPQVEASSPR
GAFGEATLYR, LGDYGLAK, LQGLVLK, VASEAPLEHKPQVEASSPR, VLACGLNEFNK	GAFGEATLYR, LGDYGLAK, VLACGLNEFNK
AGGGAAEQEELHYIPIR, S∼STVTEAPIAVVTSR, VASEAPLEHKPQVEASS∼PR, VTLLNAPTK	AGGGAAEQEELHYIPIR, VASEAPLEHKPQVEASS∼PR
	LGINLLGGPLGGK, LLTFGCNK, QLSFYK, TFDATNPLNLCVK, TLNIFLTK
	S∼STVTEAPIAVVTSR, VTLLNAPTK

The second example shows the protein CDK2. The analysis of the function plot of the partition coefficient's(Figure 5.5 middle, left hand side) results in proposing two clusters. Indeed, the discrepancies of the second, third and the fourth point are marginal. Hence, the probability for the existence of one, two and three clusters is nearly the same. Interpretation of the partition entropy (Figure 5.5 middle, right hand side) is difficult as three clusters seem to be valid, but four clusters could be possible as well. Subjective analysis of the plot of the objective function results in $c_{opt} = 4$, since this is the last significant decrease.

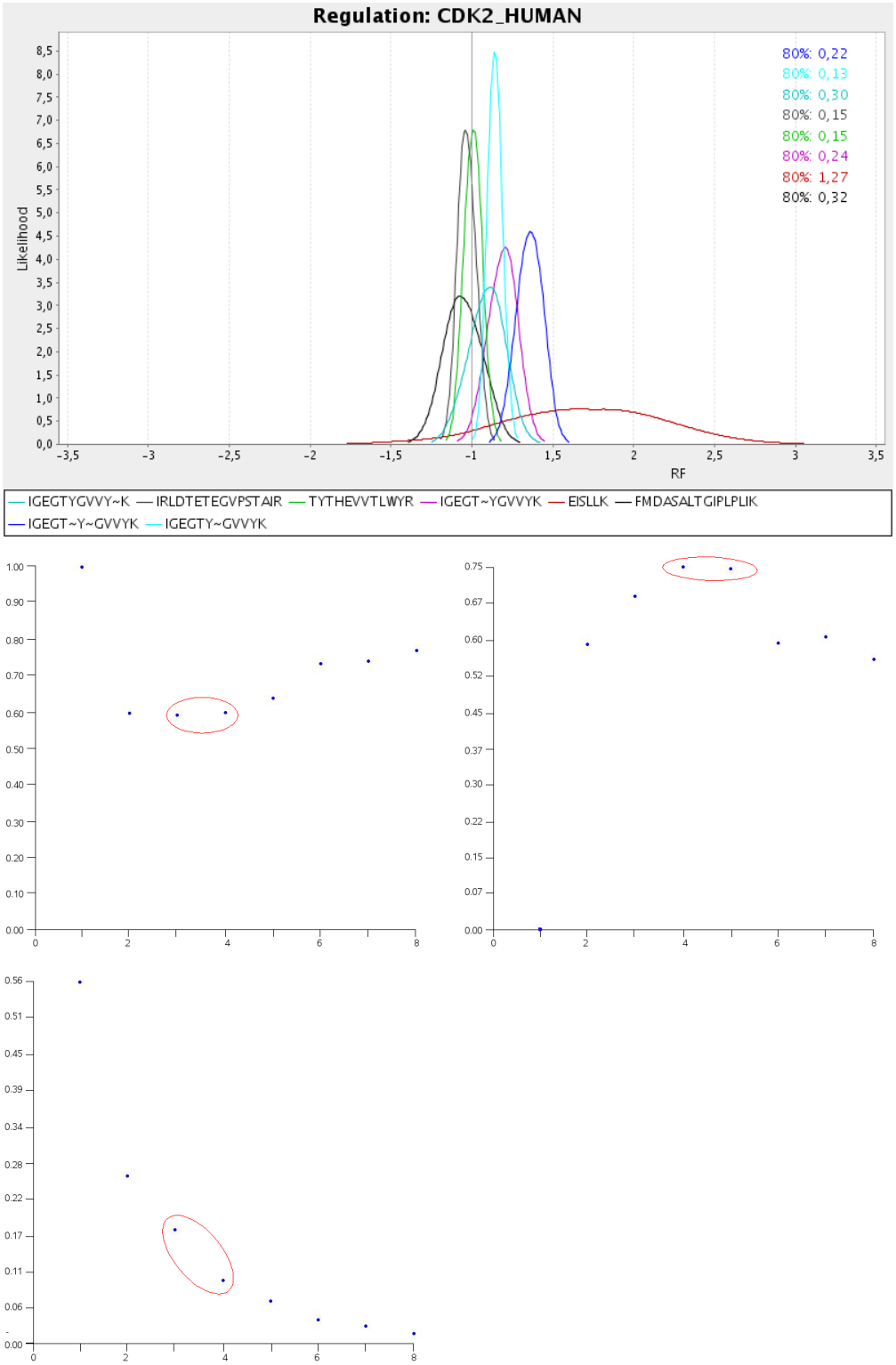


Figure 5.5: Fuzzy clustering results: Likelihood plot and validity measures of the protein CDK2. Top: Likelihood plot presenting 8 peptides clustering into 2 or 3 clusters. Middle: Function plot of the partition coefficient (left) and the partition entropy (right) indicate the existence of 2 and 3 clusters, respectively. Bottom: Objective function indicating the existence of 4 clusters.

Since the visual evaluation of validity measure plots turns out to be difficult and the results are often ambiguous, it is assumed that automatic evaluation will be difficult. Furthermore, none of the measures provides reliable results at all times and identification of proteins consisting of one cluster is not possible.

5.3.2.3 Identifying the Number of Clusters

Using the validity measures partition coefficient, partition entropy and objective function the prototype based fuzzy clustering approach does not lead to satisfying partitions of peptide likelihood curves. Therefore, other approaches for identifying the optimal number of clusters are compared in the following. Once the best fitting number is known, fuzzy clustering may find out the best partition afterwards.

Removing the most distant curves subsequently

This approach is based on the generation of a prototype for all regarded peptide likelihood curves and subsequently removing the most distant peptide curve in terms of the prototype curve. In this way the set of likelihood curves is reduced step by step and the distances of the removed curves are plotted as a plot. Analysis of entries within the plot returns the result whether there is only one cluster or not.

First of all, a distance measure d_{ij} giving the distance of two elements i (prototype i) and j (peptide j) is to be defined. In order to compare area-normalised likelihood curves i and j the size of the overlapping area A is the distance measure if there is a partial overlap of curves. Then the distance d_{ij} is given by

$$d_{ij} = \begin{cases} 0 \leq 1 - A < 1 & \text{partial overlap of the curves } i \text{ and } j \\ 0 & \text{total overlap of the curves } i \text{ and } j \end{cases} \tag{5.11}$$

with $A =$ size of overlapping areas of the curves i and j.

Contrary, if the curves i, j are not overlapping, d_{ij} is defined by the distance of the curves i and j with regard to the scaling of the x-axis which is given by the distance of the highest calculated regulation factor x_{max} of the lower regulated element el_1 and the lowest calculated regulation factor x_{min} of the higher regulated element el_2. Since $d_{ij} = 1$ if the curves i and j do not overlap, this is the minimum value for non-overlapping curves and must be added to the calculated distance. Therefore, in the case of non-overlapping curves i and j the distance measure d_{ij} is given by

$$d_{i,j} = x_{min}^{(el_2)} - x_{max}^{(el_1)} + 1 \tag{5.12}$$

Initially, a prototype curve representing all likelihood curves of the protein is calculated. In contrast to the prototype generation given in section 5.3.2.1 the distance measure d_{ij} now is based on the number of overlapping areas. Therefore, the prototype curve must overlap with the majority of the peptide likelihood curves and its likelihood has to be greater than zero at every discretised regulation factor. In detail, the prototype is calculated by the detection of the number of overlapping curves for every area that is located below the likelihood curves. Afterwards, areas, which are part of the highest number of likelihood curves are added to the prototype curve subsequently. This process is finished as soon as the total area of the prototype curve reaches 1.

The second step consists of repeatedly calculating the distances d_{ij} of every peptide likelihood curve and the prototype by application of (5.11) and (5.12) and removing the most distant object until there are no likelihood curves left. The respective distances of the removed curves are plotted against the number of iterations within a plot.

Finally, this plot shows the number of different regulatory peptide clusters. In the case of one cluster on the one hand the entries are arranged homogeneously and close to each other. In the case of multiple clusters on the other hand, there are groups of entries. After removing the last curve of a cluster a significant step can be observed in the plot. It should be noted that the distance of two curves within a cluster is always smaller than 1.

The following figures show the results obtained by the method of removing the most distant curves. Figure 5.6 illustrates the peptide likelihood plots of the proteins K2C6A and HSP71 as well as the associated plots visualising the results obtained by the method of removing the most distant curves. The left plot is associated with the upper protein (K2C6A) and shows a significant step indicating two well separated clusters. The plot on the right hand side is associated with the lower protein (HSP71) and shows no step indicating one single cluster. Both results are more or less in accordance with the corresponding likelihood plots, K2C6A may consist of two or three clusters. Figure 5.7, however, shows two proteins (NEK9 and MK01) whose numbers of clusters is wrongly determined by the presented approach. The left plot is associated with the upper likelihood plot (NEK9) and shows no significant decreasing step indicating one single cluster. The plot on the right hand side is associated with the lower protein plot (MK01) and suggests three clusters since the step between the first and the second point is very small.

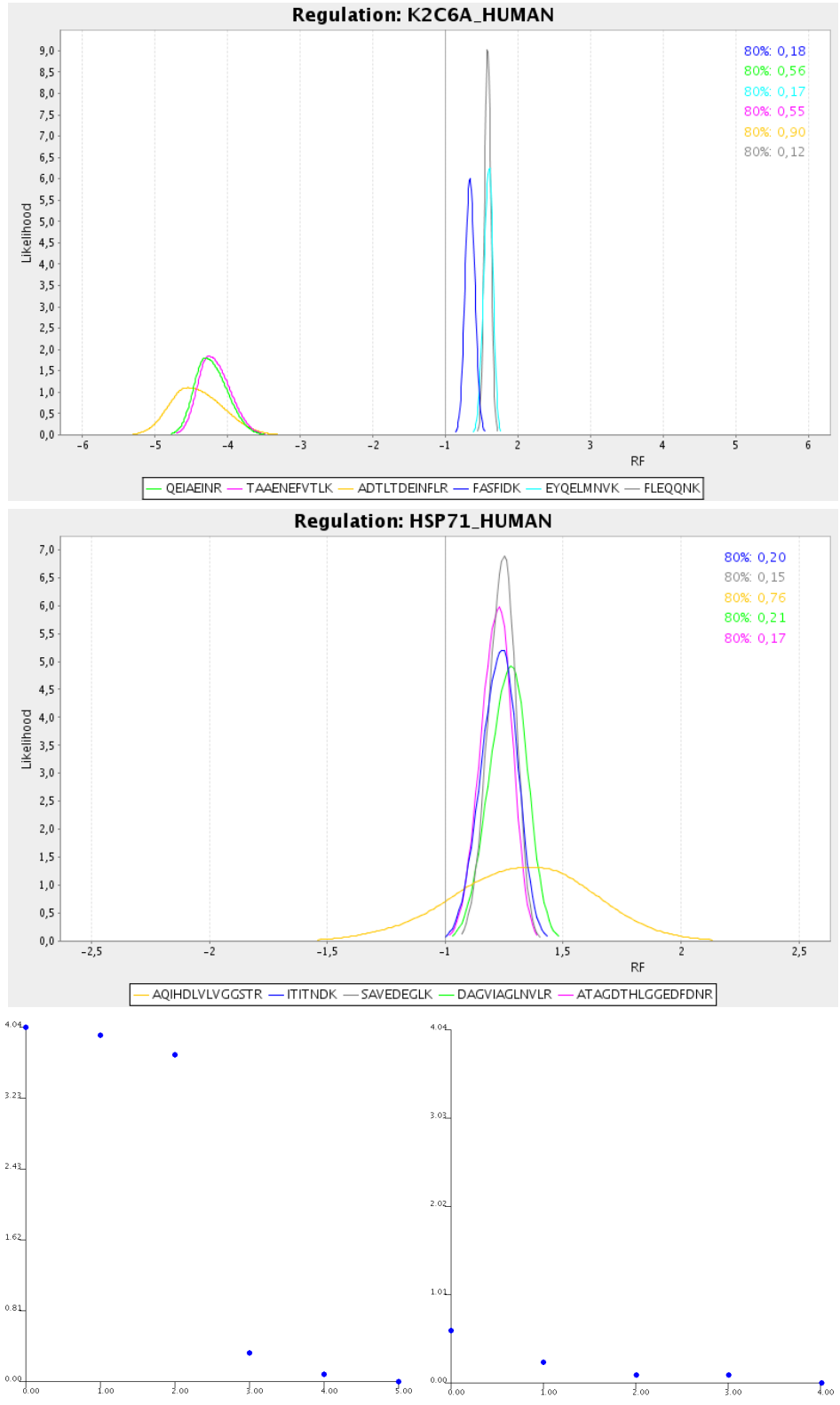


Figure 5.6: Peptide likelihood plots and corresponding function plots visualising the results obtained by the method of removing the most distant curves. The left result plot is associated with the upper protein (K2C6A) and shows a significant step indicating two well separated clusters. The plot on the right hand side is associated with the lower protein (HSP71) and shows no step indicating one single cluster.

In summary, the method returns roughly correct outcomes in the examples given in Figure 5.6, whereas it fails in the examples given in Figure 5.7. A strong limitation of this approach becomes clear when a protein is lacking a significant number of equally regulated peptides clustering into one cluster as shown in Figure 5.7.

Furthermore, the method is not able to distinguish between different clusters having similar distances to the main cluster but are placed on opposite orientations. Since the test is using distances that neglect the orientation (lower or higher regulated than the main cluster), two similar distant but adversely located clusters result in a common step in the plot. Therefore, the test is not suitable for determining the number of clusters, but only for the discrimination of one cluster on the one hand and multiple clusters on the other hand. An example is given in Figure 5.8: Next to the main cluster in the middle are additional clusters on the left as well as on the right. Both marginal clusters are located very closely to the main cluster but they are not overlapping. Regarding the corresponding result plot, it can be clearly observed that four curves are removed in the beginning and after a significant drop the remaining curves follow. The curves removed initially originate from the marginal clusters (the left cluster consists of three curves which can not be kept apart visually in the likelihood plot).

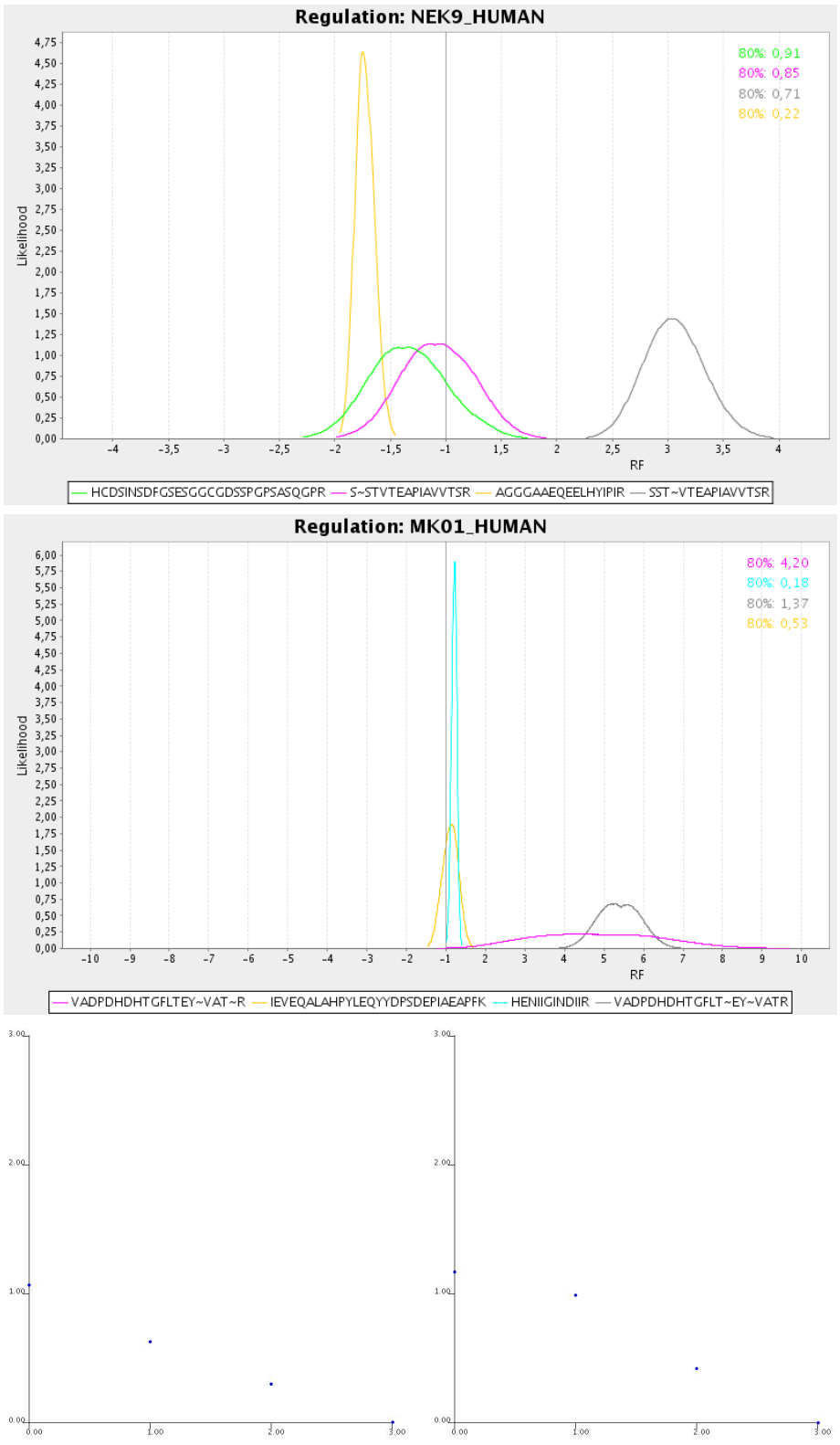


Figure 5.7: Peptide likelihood plots and corresponding function plots visualising the results obtained by the method of removing the most distant curves. The left result plot is associated with the upper protein (NEK9) and shows no significant step indicating one cluster. The plot on the right hand side is associated with the lower protein (MK01) and contrary to expectations it shows no significant step as well. Both results are not in accordance with the corresponding likelihood plots. NEK9 consists of three clusters, whereas MK01 consists of two clusters.

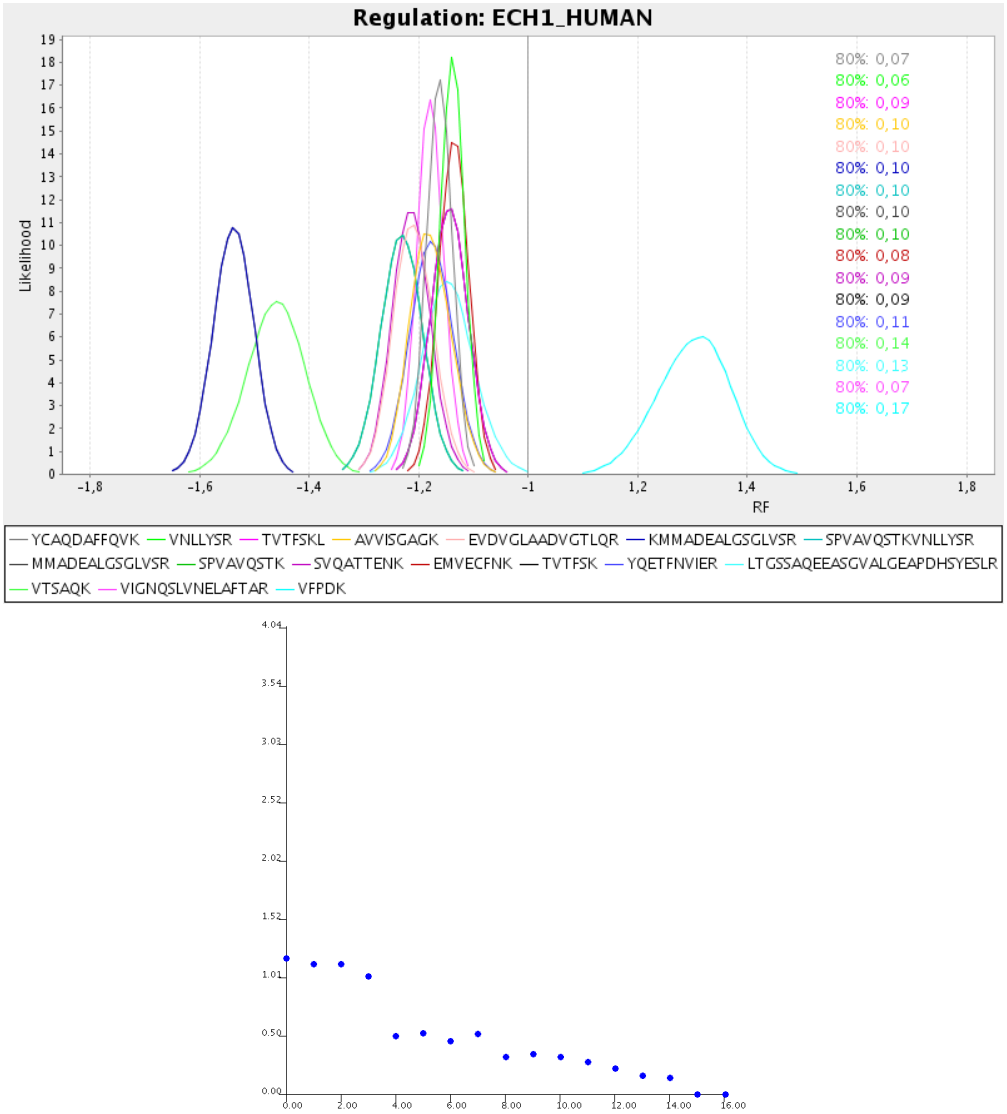


Figure 5.8: Peptide likelihood plot and corresponding function plot visualising the results obtained by the method of removing the most distant curves. The likelihood plot shows 17 peptide curves, which cluster into three groups. The leftmost cluster is composed of three curves, the cluster in the middle consists of 13 peptide curves and the rightmost cluster contains one curve. Distances between the main cluster in the middle and both marginal clusters are more or less equal. The corresponding result plot indicates that the protein is composed of two peptide clusters.

Maximisation of the Overlapping Areas

Although the former strategy is able to distinguish whether a protein consists of one or more peptide clusters, the determination of the exact number of clusters is not possible in all cases. Therefore, a new approach based on maximisation of the overlapping areas below the peptide likelihood curves is developed. The general idea is the step-by-step elimination of those curves, contributing fewest to the overlapping area A_i, in which the majority of the curves participate (i = number of contributing curves). Initially, the area A_n (n = total number of curves) that all curves participate is determined. If there is no overlapping area below all curves, the number of contributing curves is reduced. Non-overlapping curves are the first ones to be removed sequentially. The size S_i of the calculated area A_i is plotted over the number of iteration within a result plot. Subsequently, the curve which contributes least to the overlapping area is removed. However, after removing one curve from the dataset, calculation of the area where most of the curves overlap is repeated in order to remove the fewest contributing curve in the next step. Having found the actual overlapping area A_{i-1} related to the reduced dataset containing $i-1$ curves the new area size S_{i-1} is plotted. Since the total area below each likelihood curve is normalised to 1 the size of the overlapping area S is the same for all contributing curves with $0 < S \leq 1$.

The number of clusters within a protein is identified by the interpretation of the corresponding result plot. Significant steps indicate that the last curve of a distant cluster is removed and a closer cluster (or the main cluster) is touched. Even though one problem of the previous strategy – lack of a significant number of equally regulated peptides – is resolved, the second issue remains. The main problem of this new approach is to unify further clusters besides the main cluster, if all clusters are well separated. In this case, the result plot only shows one significant step between the main cluster and a combination of the further clusters. Therefore, the method of maximisation of overlapping areas is helpful concerning the detection of proteins clustering into one single cluster. For the determination of the optimal number of clusters, however, this strategy fails. Figure 5.9 gives two examples. The upper peptide plot shows a protein consisting of four peptides clustering into two clusters. As the clusters overlap, the presented approach is able to detect both clusters, which can be identified by the step between the second and the third point in the result plot on the left hand side. The lower likelihood plot shows a protein consisting of 17 peptides clustering into three groups. The leftmost as well as the rightmost cluster do not overlap with the main cluster in the middle. In the result plot, the leftmost four points seem to belong to one cluster which is not the case. In fact, these four points belong to both the right

cluster consisting of one curve and to the left cluster consisting of three curves (two of them can not be distinguished since they are identical). Regarding the second example this strategy is also not able to determine the optimal number of peptide clusters of the protein.

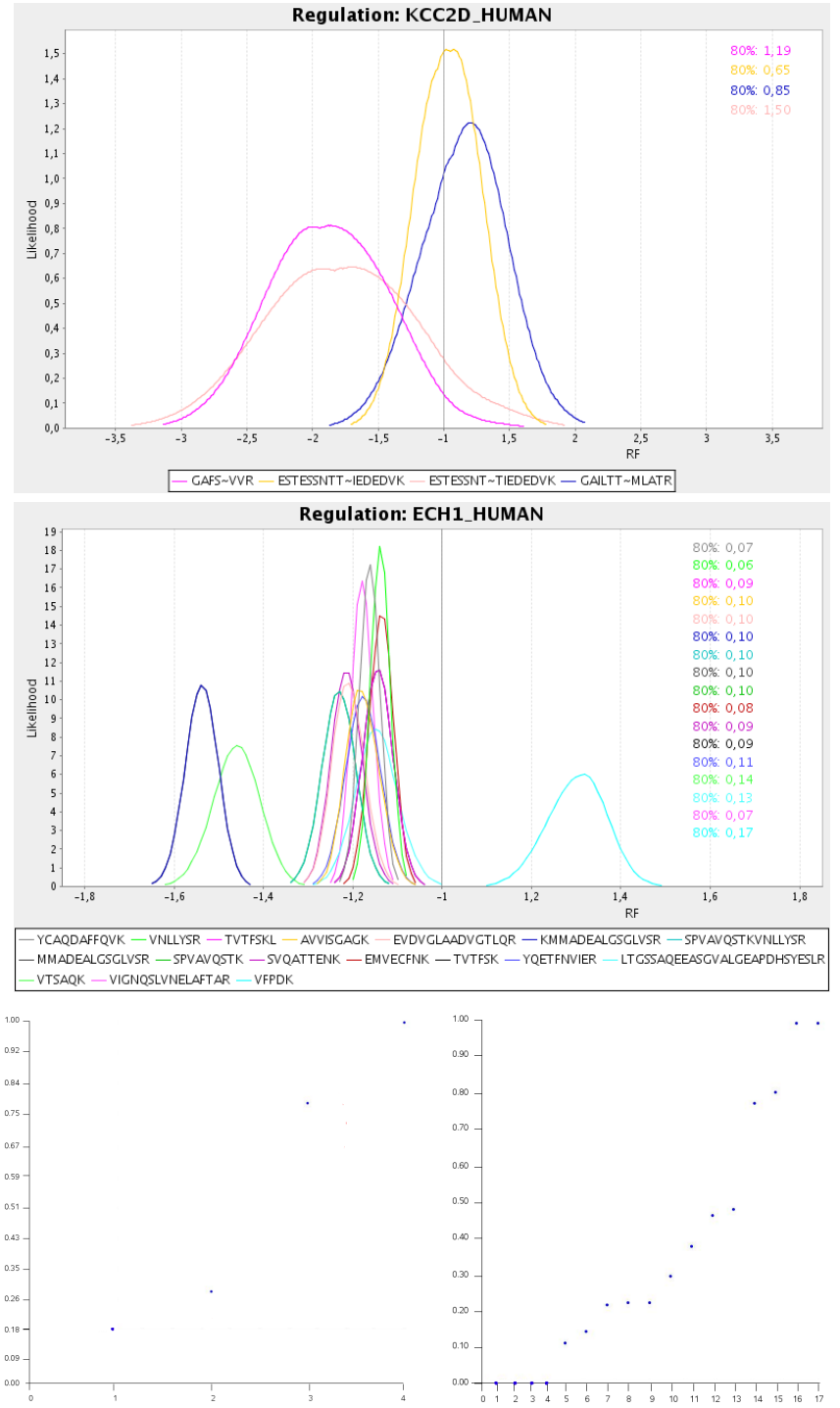


Figure 5.9: Peptide likelihood plots and corresponding result plots visualising the outcomes obtained by the method of removing least overlapping curves. The upper likelihood plot shows a protein consisting of four peptides clustering into two clusters. The presented approach is able to detect both clusters, which can be identified by the step between the second and the third point in the result plot on the left hand side. The lower likelihood plot shows a protein consisting of 17 peptides that cluster into three groups. The leftmost as well as the rightmost cluster do not overlap with the main cluster in the middle. In the result plot the leftmost four points seem to belong to one cluster which is not the case. In fact, these four points belong to the right cluster consisting of one curve and to the left cluster consisting of three curves (two of them can not be distinguished since they are identical).

Maximum Fitness

The basis of the former approaches are founded on similarity between overlapping areas below the likelihood curves. Both strategies are able to decide whether a protein consists of one or more peptide clusters, but due to different resons, they are not able to determine the number of clusters exactly. Thus, a further approach is presented which is based on maximising the product of the likelihoods of all peptides within in the same cluster. This method allows the detection of outlying curves by concerning likelihoods instead of areas below cuves. Thereby, every peptide curve is compared separately with all other curves of the protein. The optimal overall regulation factor is found by maximising the total likelihood by calculation of L_n for every possible regulation factor by

$$L_n = \prod_{i=1}^{n} l_{\mathrm{rf}}^{(i)} \tag{5.13}$$

where rf is the regarded regulation factor, $l_{\mathrm{rf}}^{(i)}$ = likelihood of curve i at regulation factor rf and n = number of peptide curves. Adding a constant value (e.g. 0.1) on every likelihood avoids to obtain zero for the whole result whenever one of the factors is zero. This is similar to so-called laplace correction as commonly used for naive bayes classifications for instance.

Following, every peptide curve x is removed individually and according to (5.13) the total likelihood of the remaining peptides L_{n-x} is calculated which is based on the optimal overall regulation factor relating to the remaining peptides. The optimal regulation factor rf_x and the corresponding likelihood $L_x = l_x$ of the removed curve are given by the x- and the y-coordinate of the maximum peak. From these values a weighted sum $L_{n,x}$ is produced by

$$L_{n,x} = \frac{n-1}{n} L_{n-x} + \frac{1}{n} L_x. \tag{5.14}$$

Subsequently, the coefficient f_x is calculated for every removed curve x by

$$f_x = \frac{L_{n,x}}{L_n} \tag{5.15}$$

In conclusion, with increasing distance of the removed peptide curve x and the remaining curves the coefficient f_x is growing. If the removed curve is located within the main cluster, f_x is low. For the analysis of results generated this way, coefficients are to be found, that are significantly higher than those derived from the other peptides. Unfortunately, the limits of this approach are met, if the number of equally regulated peptides is not significantly higher than the number

of differentially regulated peptides. Then the basis for the outlier recognition is missing. The following figures (Figure 5.10 and Figure 5.11) illustrate the results of this approach based on maximising the likelihoods. Both figures show two peptide likelihood plots and the corresponding results of the maximum fitness method visualised by means of result plots depicting the coefficients f_x after removing curve x. The respective coefficient of the removed curve is plotted against the number of the curve. The approach successively analyses the peptides of the proteins CSK21 and CDC2 (Figure 5.10). The resulting plot on the left hand side related to the upper protein clearly shows one significant outlier and the plot on the right hand side related to the lower protein shows three outliers. Figure 5.11 gives two examples which illustrate cases in that the maximum fitness method fails. Both proteins consist of several clusters and lack a main cluster which serves as base containing a large part of the peptides curves. Hence, no significant outliers can be found in the resulting plots related with the proteins KC1D (left hand side) and CDC2 (right hand side).

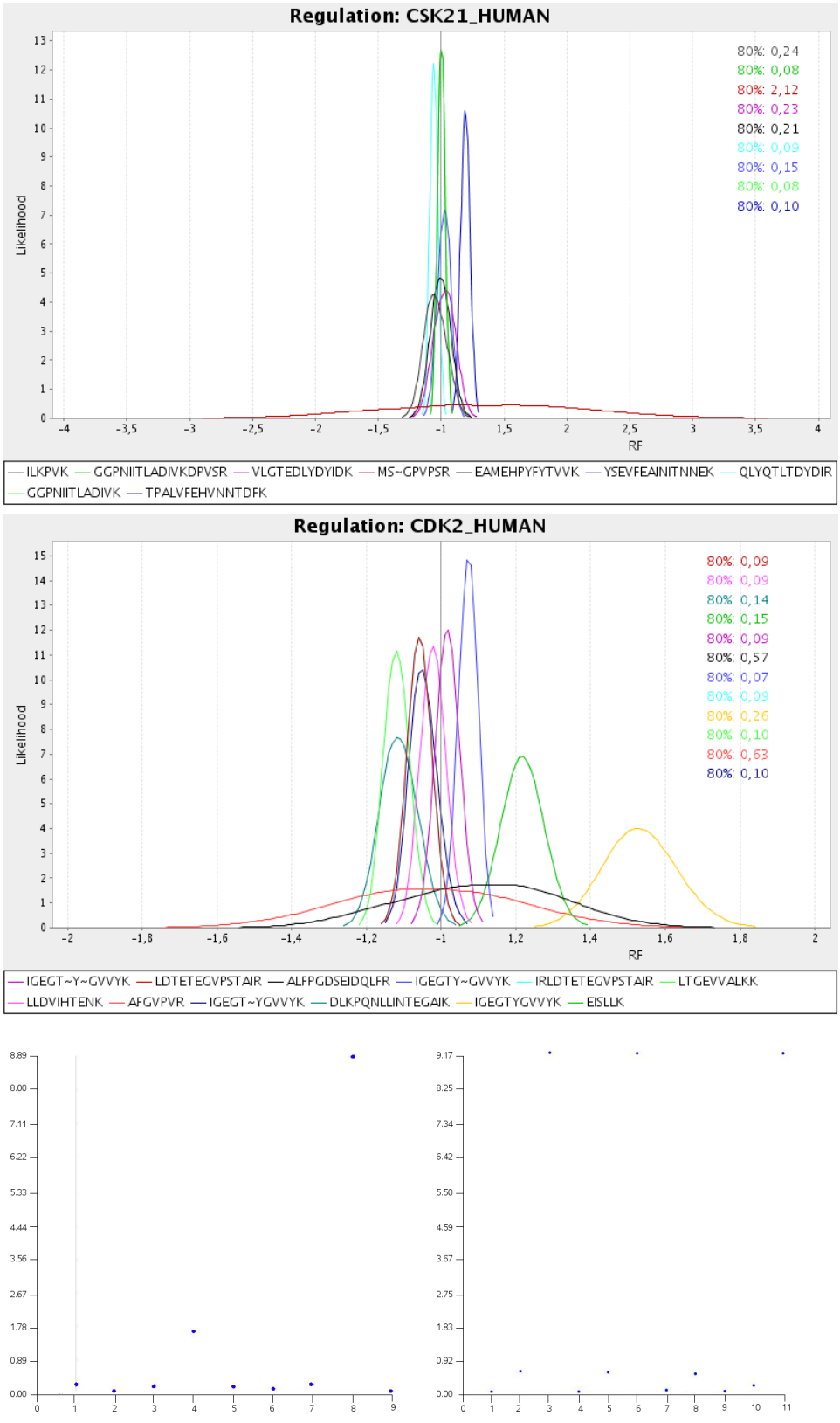


Figure 5.10: Likelihood plots and corresponding result plots visualising the outcome of the maximum fitness method for the identification of outlying peptides within one protein. The upper protein is associated with the left result plot, the lower protein is associated with the right one. Both resulting plots are based on a high number of curves building the main cluster, one (CSK21) and three (CDK2) significant outliers can be distinguished.

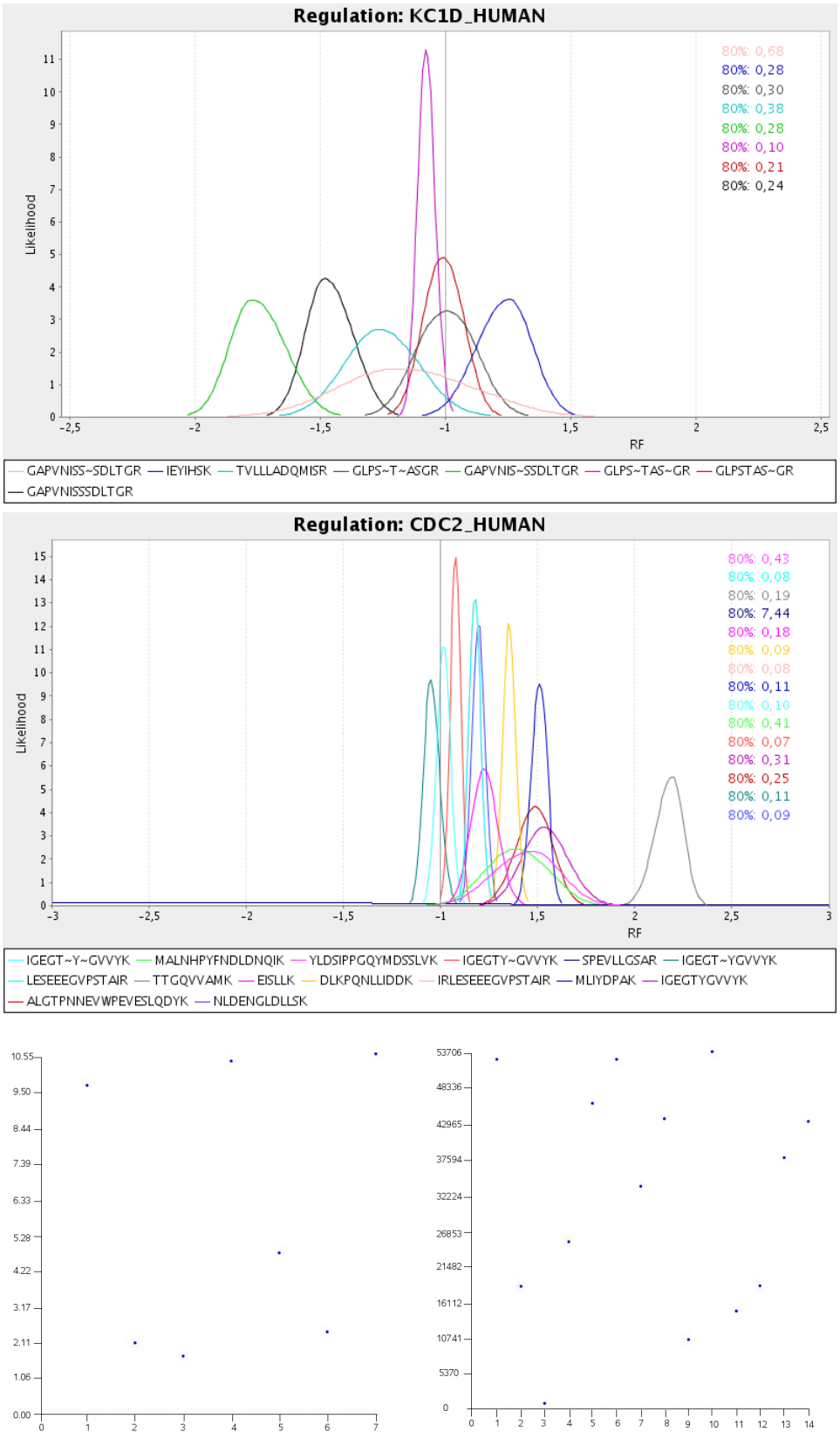


Figure 5.11: Likelihood plots and corresponding result plots visualising the outcome of the maximum fitness method for the identification of outlying peptides within one protein. The upper protein is associated with the left result plot, the lower one is associated with the right plot. Both resulting plots show no clear outliers due to the lack of a main cluster serving as data basis.

5.3.3 Expectation-Maximisation Clustering

In summary, none of the strategies presented in section 5.3.2.3 is able to determine credibly the exact number of peptide clusters. Each of them fails because of at least one of the reasons: (i) lacking a main cluster of similar regulated peptide curves that serve as basis (ii) failure to dinstinguish clusters regulated with similar regulations in different directions in comparison with the main cluster.

So far, peptide likelihood curves were regarded as abstract objects and standard crisp or fuzzy clustering algorithms were applied after adaption to the underlying data. Most of the approaches for the development of validity measures focused on maximising the number of overlapping curves and the determination of distances and similarities.

However, in the purpose of the noise model likelihoods are to be maximised. Therefore, an expectation-maximisation (EM) clustering algorithm is designed for finding clusters based on maximisation of the likelihoods. This approach belongs to the type of partitioning clustering algorithms.

5.3.3.1 Introduction into Expectation-Maximisation Clustering

The EM clustering algorithm computes probabilities of cluster memberships based on one or more probability distributions. The goal of the clustering algorithm then is to maximise the overall probability or likelihood of the data, given the (final) clusters (Dempster *et al.*, 1977; Hill and Lewicki, 2006).

5.3.3.2 Expectation-Maximisation Clustering of Peptide Likelihood Curves

This perferctly probabilistic approach acts on assumptions concerning both the number of regulated peptides and the noise model. The basic idea is to find j optimal regulation factors $\mathrm{rf}_j, 1 \leq j \leq n$, that each represent one cluster and are able to generate i related likelihood curves, $1 \leq i \leq k$. Every curve is related with that regulation factor which is the most likely resulting in the highest likelihood. The total likelihood L_j for any cluster j is calculated by

$$L_j = \prod_{i=1}^{k} f_{i,\mathrm{rf}_j} \tag{5.16}$$

with k = number of curves related to cluster j and $f_{k,\mathrm{rf}}$ = likelihood (fitness) of curve i at regulation factor rf.

The optimal regulation factor of each cluster j is found by calculation of the maximum L_j for every possible regulation factor within the cluster. Two important factors have a significant impact on the assignment of curves to one cluster: First of all, the initial partition strongly depends on the initialisation of regulation

factors representing the specific clusters. Initialisation is performed by randomly choosing j optimal peptide regulation factors which are given by the peaks of the likelihood curves. This process is repeated in proportion to the number of peptide curves. A frequent problem of this method is that single outlying curves for the initialisation step can be missing. For example, a protein consisting of 20 peptides, from which 18 peptides cluster into two clusters and two peptides are each separated, will be rarely initialised with both separated curves. As a consequence, they will be related to other clusters instead of forming single clusters. For avoiding these cases, the first of the repeated initialisation steps is predetermined: Since outlier rarely are located in the centre of the plot, the curves which are used for initialisation are those, having the lowest and the highest most likely regulation factors. Following the initialisation, the partition of the clusters is modified – which is the second important factor. As long as an improvement of the final result is achieved, one curve is removed from its cluster and is related to another one. For this reassignment, one curve has to be found, whose likelihood contribution to a different cluster, that the curve is not assigned to, is highest. After removing the selected curve from its previous cluster and adding it to the second best cluster, the optimal regulation factors for all clusters are recalculated by application of (5.16) and following maximisation of L_j.

Once the optimal regulation factors rf_j for all clusters j are found and the cluster likelihoods are computed, the total likelihood L_c is calculated by

$$L_c = P_c(\sum_{j=1}^{c} \frac{k}{n} \prod_{i=1}^{k} l_{i,\mathrm{rf}_j}) \tag{5.17}$$

with P_c = probability of a protein clustering into c clusters according to a-priori distribution P, c = number of clusters, k = number of curves related to cluster j, n = total number of peptide curves within the protein and l_{i,rf_j} = likelihood of curve i for regulation factor rf_j.

Finally, the highest of the maximum total likelihoods L_c for every possible number of clusters c gives the optimal cluster number as well as the optimal cluster partition. The a-priori distribution P gives an instrument for weighting the possible numbers of clusters whereby expert knowledge about the frequency of typical cluster numbers can be introduced. Table 5.2 gives the a-priori distribution, that is used for the calculation of the results in this work.

5.3.3.3 Results of Expectation Maximisation Clustering

All examples given in the following are performed by using an a-priori distribution P_c in (5.17) allowing the introduction of expert knowledge. From experiments it

is estimated that the number of clusters averages the following frequencies (Table 5.2):

Table 5.2: A-priori distribution of occurring number of clusters in common protein datasets derived from experiments.

1 Cluster	2 Clusters	3 Clusters	4 Clusters	5 Clusters	6 Clusters	≥ 7 Clusters
0.25	0.45	0.15	0.08	0.04	0.02	$\sum_{k=7}^{n} q_k = 0.01$

Several results returned by the previously discussed methods are difficult to evaluate, mainly due to the determination of cluster numbers (i.e. Figures 5.4, 5.5, 5.8, 5.9, 5.11). In contrast, expectation maximisation clustering determines several cluster partitions according to the strategy presented in section 5.3.3 and returns that one which achieved the highest total likelihood. Hereby, the quality of the analysis strongly depends on initialisation and variation of the assignment of peptide curves to clusters. The advantage of this EM clustering approach is clearness of its result: The total likelihood (numerical data) has to mximised and can be compared for all possible numbers and partitions of clusters.

If a precise result can be achieved by one of the previous presented approaches, it is often similar to that, which is returned by expectation maximisation clustering. The most problems concerning the determination of the number or the partition of clusters were observed in those cases when a protein is identified only by a few peptides or when its peptides belong to several clusters which are regulated differently. A clear result is found by all presented algorithms if a protein is identified by several similar regulated peptides and one outlying peptide. In contrast, all of the previous methods fail if the number of consistently regulated peptides is low (e.g. 2) and the same number of further peptides (2) is differently regulated as given in Figure 5.12. However, here the expectation maximisation clustering clearly results in three clusters: one cluster containing the likelihood curves of the peptides `HCDSINSDFGSESGGCGDSSPGPSASQGPR` (green) and `AGGGAAEQEELHYIPIR` (yellow), another cluster consisting of `S`∼`STVTEAPIAVVTSR` (purple) and a third cluster containing `SST`∼`VTEAPIAVVTSR` (grey).

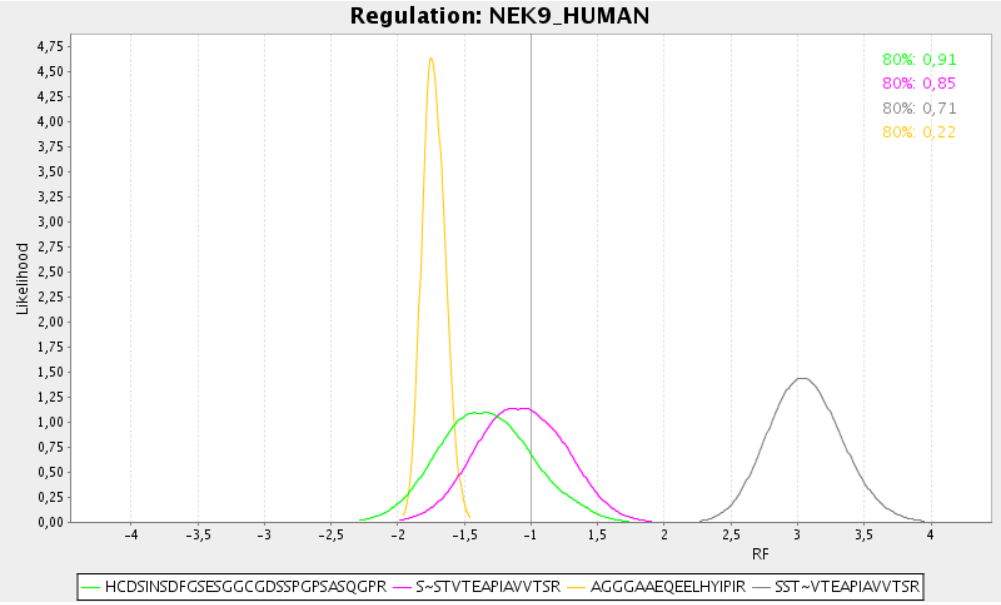


Figure 5.12: Protein NEK9 identified by four peptides clustering into three clusters. Both leftmost peptide likelihood curves are assigned to the same cluster (yellow, green). Single peptide clusters are each detected containing one nearly unregulated curve (purple) and a significantly upregulated curve (grey).

HSP71 (Figure 5.13) was identified by five unphosphorylated peptides. The EM clustering algorithm identifies only one cluster.

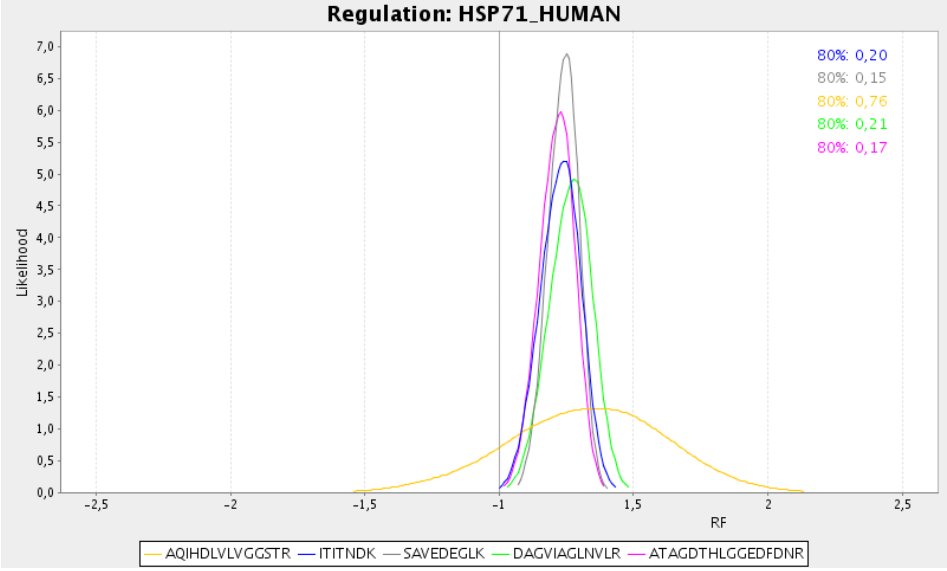


Figure 5.13: Likelihood plot of the protein HSP71.

Limitations of EM Clustering

Multiplication of the peptide likelihoods for the calculation of the cluster likelihood in (5.16) causes that forming a low number of clusters is preferred by the algorithm. The increase of the total likelihood (compare (5.17)) is significantly higher when an additional curves is added to the cluster resulting in multiplying the former cluster likelihood by the likelihood of the additional curve. This is true if the additional likelihood is greater than 1. In comparison, forming a separate cluster causes a minimal increase of the total likelihood.

Furthermore, due to the weighting of the determined clusters by $\frac{k}{n}$ with $k =$ number of curves within the cluster and $n =$ number of total curves of the protein the algorithm tends to assign low-quality likelihood curves to clusters containing a little number of curves which sometimes hides the existence of clusters consisting of only one single curve.

For avoiding these effects, it is very profitable to remove well separated single peptide clusters before applying the EM clustering algorithm. An intuitive criterion for the identification of those is the calculation of overlaps of curves. If one curve does not overlap with other curves with more than 5% of its total area, it is defined to be well separated and removed from the dataset containing the peptide likelihoods. Examples are given in Figure 5.14 and Table 5.3.

NEK9 (Figure 5.14, Table 5.3) was identified by 17 peptides that are assigned to seven clusters by the EM clustering algorithm. NEK9 is known to be phosphorylated at S_{332} and T_{333} (peptide `SSTVTEAPIAVVTSR`).

EM clustering itself identifies only four peptide clusters within the data of NEK9 in the case of application without removing well separated single peptide clusters. In this result two of the upregulated phosphopeptides are assigned to other clusters as well as the corresponding downregulated unmodified peptide. Application of EM clustering after removing clear single peptide clusters yields a very good result identifying the different regulation of the unmodified peptide corresponding to the phosphopeptides.

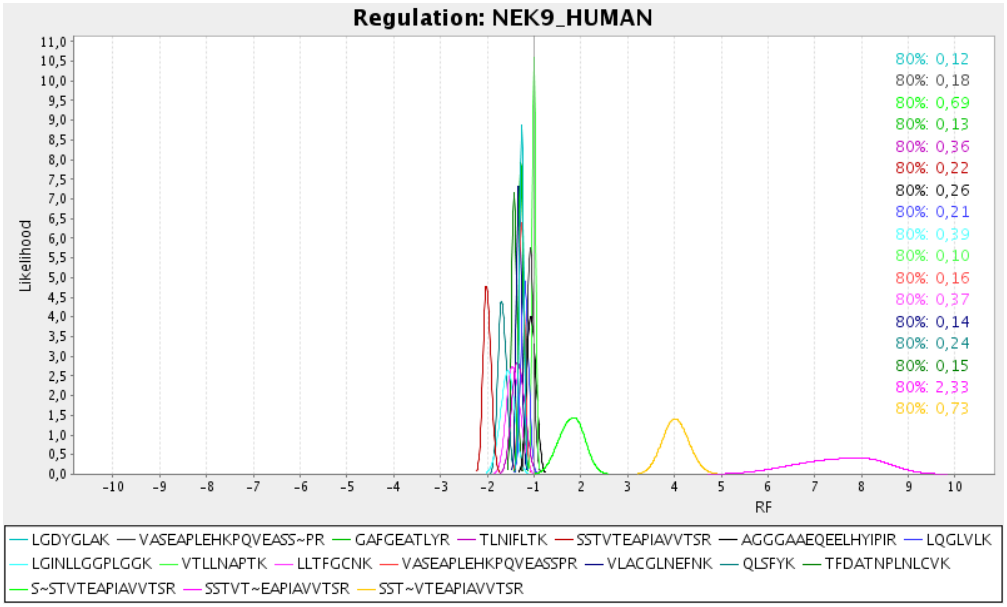


Figure 5.14: Likelihood plot of NEK9 clustering into at least five clusters.

Table 5.3: Partition of peptides of the protein NEK9 into seven clusters which were calculated by the presented EM clustering algorithm.

Cluster	Peptides (RF)
1	SSTVTEAPIAVVTSR (−2.02)
2	QLSFYK (−1.7), LGINLLGGPLGGK (−1.56)
3	LLTFGCNK (−1.47), TFDATNPLNLCVK (−1.43), TLNIFLTK (−1.36), VLACGLNEFNK (−1.34), GAFGEATLYR (−1.29), VASEAPLEHKPQVEASSPR (−1.28), LGDYGLAK (−1.27), LQGLVLK (−1.22)
4	VASEAPLEHKPQVEASS~PR (−1.08), AGGGAAEQEELHYIPIR (−1.07), VTLLNAPTK (1.0)
5	S~STVTEAPIAVVTSR (1.84)
6	SST~VTEAPIAVVTSR (4.02)
7	SSTVT~EAPIAVVTSR (7.90)

MK01 (Figure 5.15, Table 5.4) was identified by three upregulated phosphopeptides (identical sequences, different phosphosites), six slightly downregulated unmodified peptides and the unmodified peptide corresponding to the phosphopeptides which is significantly downregulated. Eight clusters are determined by EM clustering. Indeed, two phosphopeptides (RF 4.50 and 4.69) obviously forming one cluster (similar regulations and shapes) are assigned to separate clusters. This is resulting from maximum likelihoods < 1 leading to a low result during mulitplication. On the other hand, addition of separate clusters gains the sum of the maximum likelihoods.

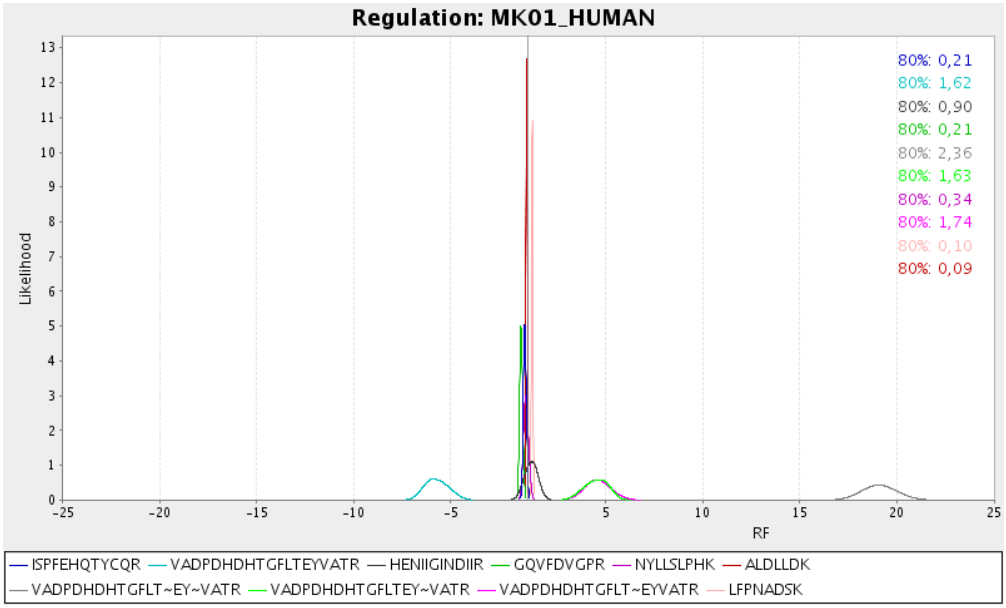


Figure 5.15: Likelihood plot of MK01 forming at least four clusters.

Table 5.4: Partition of peptides of the protein MK01 into eight clusters which were calculated by the presented EM clustering algorithm.

Cluster	Peptides (RF)
1	VADPDHDHTGFLTEYVATR (−5.93)
2	GQVFDVGPR (−1.40)
3	ISPFEHQTYCQR (−1.20), NYLLSLPHK (−1.15), ALDLLDK (−1.08)
4	HENIIGINDIIR (1.13)
5	LFPNADSK (1.20)
6	VADPDHDHTGFLT∼EYVATR (4.50)
7	VADPDHDHTGFLTEY∼VATR (4.69)
8	VADPDHDHTGFLT∼EY∼VATR (18.91)

Results

In the following examples the EM clustering approach is applied to the proteins shown in the previous examples. Most of them could not be processed correctly by the former clustering approaches whereas EM clustering returns clear and plausible results. Figures 5.16 - 5.22 dipict the likelihood plots, Tables 5.5 - 5.11 list all identified peptides, the calculated regulation factors determined by (4.2) and the assignment of the peptides to clusters calculated by EM clustering.

All peptides forming own clusters are checked by a database query (Uniprot/Swissprot[3]) whether they can be the target of proteolysis (if they are located in the beginning or in the end of the protein), or are known to be modified. All examples are taken from the dataset F074086.dat (HGF experiment, T. Reinl).

[3]http://www.expasy.org/uniprot/

I. ECH1 (Figure 5.16, Table 5.5) was identified by 14 unphosphorylated peptides that were assigned to three clusters by EM clustering. The peptide `VFPDK` forming an own cluster is not known to be modified. Degradation can be excluded due to the localisation of this peptide (positions 60-65 from 328 amino acids in total) within the protein. Therefore, this peptide has to be analysed regarding isoforms and wrong assignment in order to decide whether a novel modification can be assumed and further investigations are started.

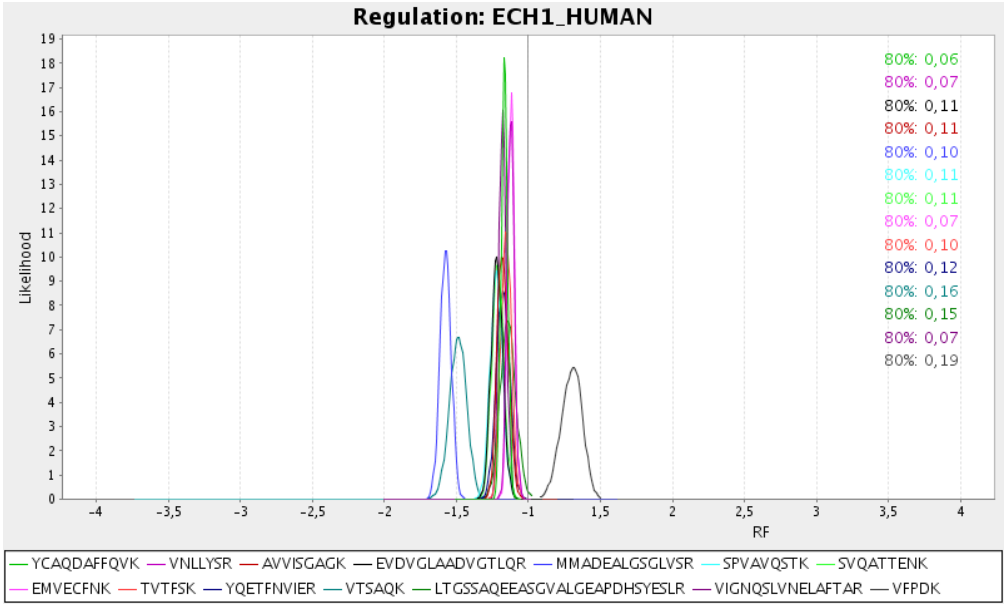


Figure 5.16: Likelihood plot of the protein ECH1 (no modifications detected).

Table 5.5: Partition of peptides of the protein ECH1 into three clusters which were calculated by the presented expectation-maximisation clustering algorithm.

Cluster	Peptides (RF)
1	(K)MMADEALGSGLVSR (−1.57), VTSAQK (−1.49)
2	EVDVGLAADVGTLQR (−1.22), SPVAVQST(KVNLLYSR) (−1.22), SVQATTENK (−1.21), AVVISGAGK (−1.19), VIGNQSLVNELAFTAR (−1.18), YQETFNVIER (−1.18), TVTFSK(L) (−1.16), YCAQDAFFQVK (−1.16), LTGSSAQEEASGVALGEAPDHSYESLR (−1.14), EMVECFNK (−1.12), VNLLYSR (−1.12)
3	VFPDK (1.32)

II. KC1D (Figure 5.17, Table 5.6) was identified by eight peptides which cluster into three clusters. Among these are two known phosphopeptides which are phosphorylated at five different amino acids. The peptide `IEYIHSK` forming an own cluster is not known to be modified. Degradation is excluded due to the localisation (positions 116-121 from 415 amino acids in total) of the peptide within the protein. This peptide has to be analysed regarding isoforms and wrong assignment in order to decide whether a novel modification can be assumed and further investigations are started.

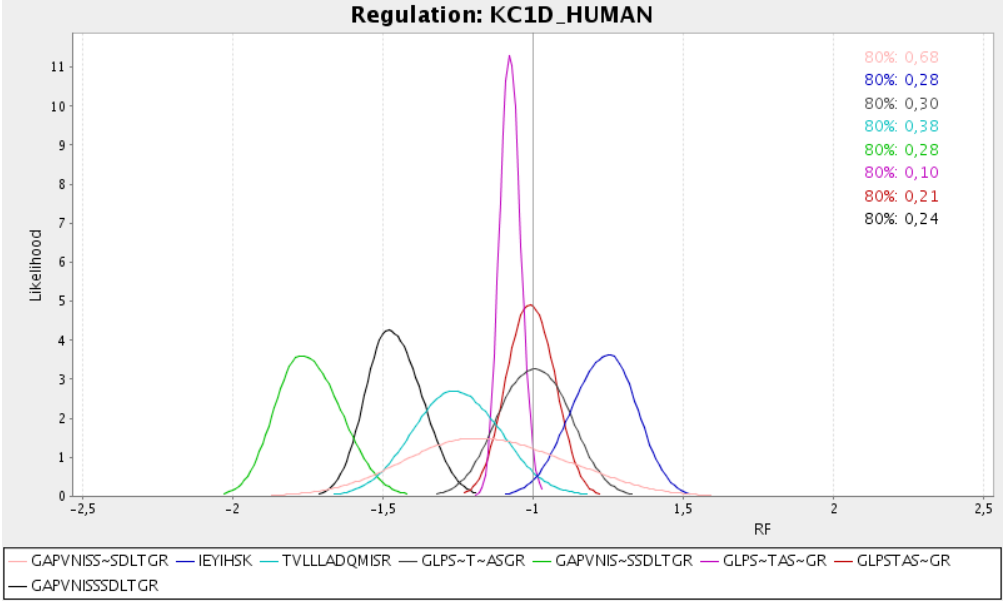


Figure 5.17: Likelihood plot of the protein KC1D, 4 phosphopeptides are detected.

Table 5.6: Partition of peptides of the protein KC1D into three clusters which were calculated by the presented expectation-maximisation clustering algorithm.

Cluster	Peptides (RF)
1	GAPVNIS∼SSDLTGR (−1.77), GAPVNISSSDLTGR (−1.48)
2	TVLLLADQMISR (−1.26), GAPVNISS∼SDLTGR (−1.17),
	GLPS∼TAS∼GR (−1.08), GLPSTAS∼GR (−1.01), GLPS∼T∼ASGR (1.01),
3	IEYIHSK (1.25)

III. K2C6A (Figure 5.18, Table 5.7) was identified by six unphosphorylated peptides which are assigned to four clusters. The peptide `FASFIDK` forming an own cluster is not known to be modified. Degradation is excluded due to the localisation (positions 174-180 from 564 amino acids in total) of the peptide in the middle of the protein. Therefore, this peptide has to be analysed regarding isoforms and wrong assignment in order to decide whether a novel modification can be assumed and further investigations are started.

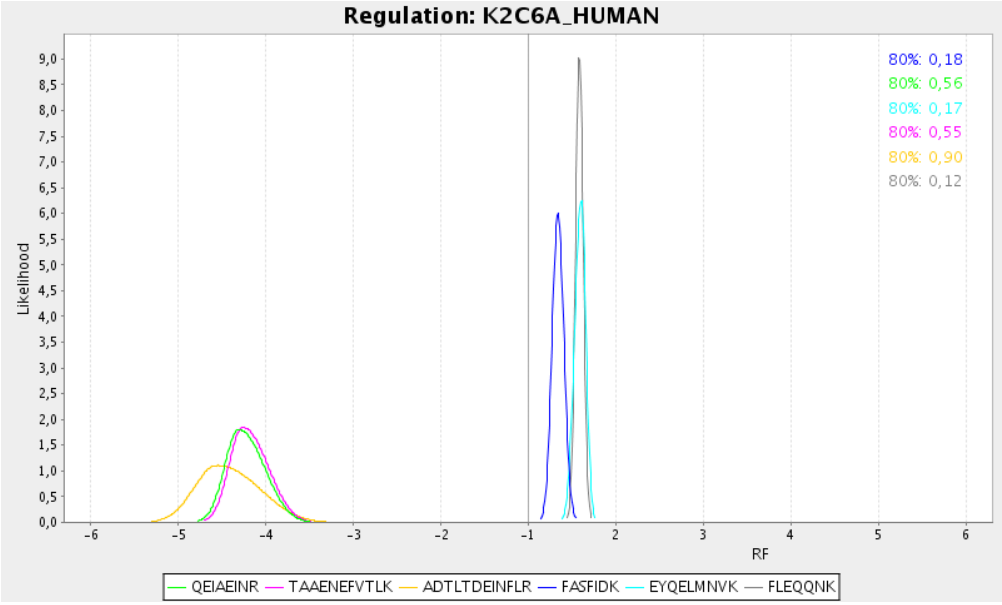


Figure 5.18: Likelihood plot of the protein K2C6A.

Table 5.7: Partition of peptides of the protein K2C6A into three clusters which were calculated by the presented expectation-maximisation clustering algorithm.

1	ADTLTDEINFLR (−4.56), QEIAEINR (−4.31), TAAENEFVTLK (−4.27)
2	FASFIDK (1.34)
3	FLEQQNK (1.58), EYQELMNVK (1.61)

IV. CDC2 (Figure 5.19, Table 5.8) was identified by 15 peptides. Among these is one known phosphopeptide which is phosphorylated at two different amino acids. EM clustering with removing the single peptide cluster `TTGQVVAMK` determines six clusters conforming the visual presentation. The database query returns no known modifications regarding the upregulated peptide `TTGQVVAMK`. However, since it is located in the beginning of the protein (positions 25-34 from 297 amino acids in total) degradation can not be excluded for causing the different regulation.

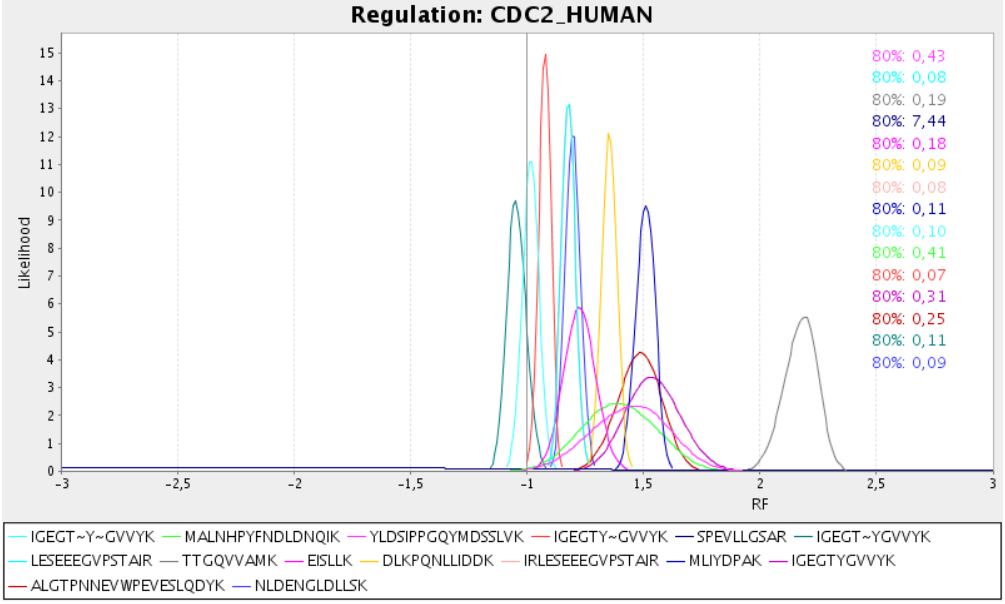


Figure 5.19: Likelihood plot of the protein CDC2 (3 phosphopeptides detected).

Table 5.8: Partition of peptides of the protein CDC2 into six clusters which were calculated by the presented expectation-maximisation clustering algorithm.

Cluster	Peptides (RF)
1	SPEVLLGSAR (−2.19)
2	IGEGT∼YGVVYK (−1.05), IGEGT∼Y∼GVVYK (1.02)
3	IGEGTY∼GVVYK (1.08)
4	(IR)LESEEEGVPSTAIR (1.18), NLDENGLDLLSK (1.19), EISLLK (1.22)
5	DLKPQNLLIDDK (1.35)
6	MALNHPYFNDLDNQIK (1.39), YLDSIPPGQYMDSSLVK (1.48), ALGTPNNEVWPEVESLQDYK (1.49), MLIYDPAK (1.52), IGEGTYGVVYK (1.53)
7	TTGQVVAMK (2.20)

V. The protein ACLY (Figure 5.20, Table 5.9) was identified by four slightly downregulated peptides and one slightly upregulated peptide, which cluster into two clusters. None of these peptides is known to be modified. The peptide `LLVGVDEK` is detected as single cluster peptide. A database query returns no known modifications concerning this peptide. Since it is located in the middle of the protein (positions 152-159 from 1101 amino acids in total) proteolysis can be excluded. Analyses regarding isoforms and wrong assignment are necessary in order to decide whether a novel modification can be assumed and further investigations are started.

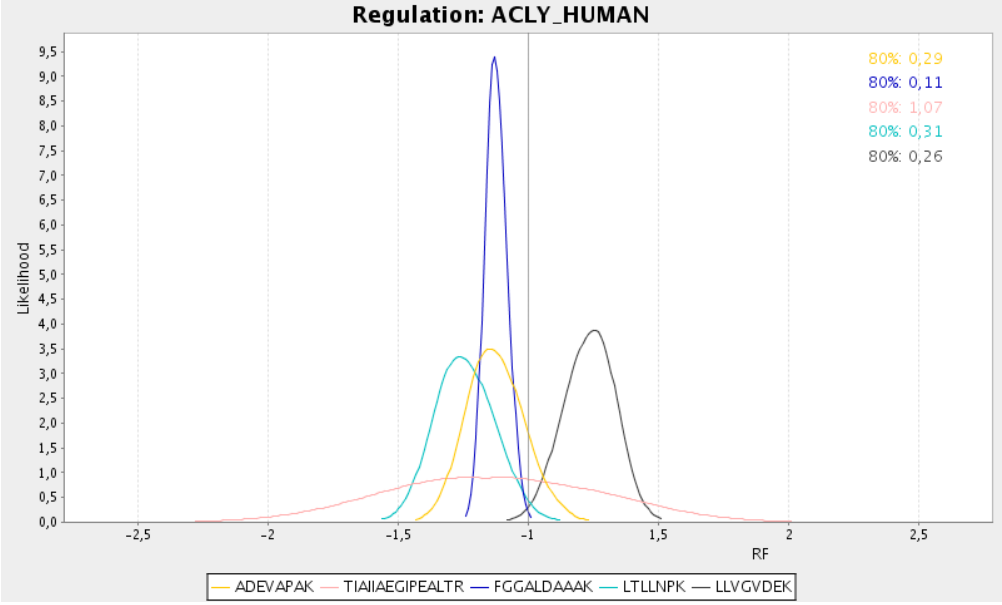


Figure 5.20: Likelihood plot of the protein ACLY.

Table 5.9: Partition of peptides of the protein ACLY into two clusters which were calculated by the presented expectation-maximisation clustering algorithm.

Cluster	Peptides (RF)
1	LTLLNPK (−1.26), ADEVAPAK (−1.14),
	FGGALDAAAK (−1.13), TIAIIAEGIPEALTR (−1.12)
2	LLVGVDEK (1.26)

VI. CCNA2 (Figure 5.21, Table 5.10) was identified by eight peptides (Figure 5.21, Table 5.10) which are assigned to three clusters. Most of the peptides are not regulated or slightly downregulated. The peptides `AALAVLK` and `YHGVSLLNPPETLNL` are identified to form own clusters. A database query returns no known modifications concerning these peptides. The amino acids of the peptide `YHGVSLLNPPETLNL` are the last ones of the protein (positions 420-432 from 432 amino acids in total). Therefore, besides a potential unknown modification protein degradation can not be excluded. Due to the location of `AALAVLK` is located in the middle of the protein (positions 152-159 from 432 amino acids in total) and hence, proteolysis can be excluded. Analyses of the latter peptide regarding isoforms and wrong assignment are necessary in order to decide whether a novel modification can be assumed and further investigations are started.

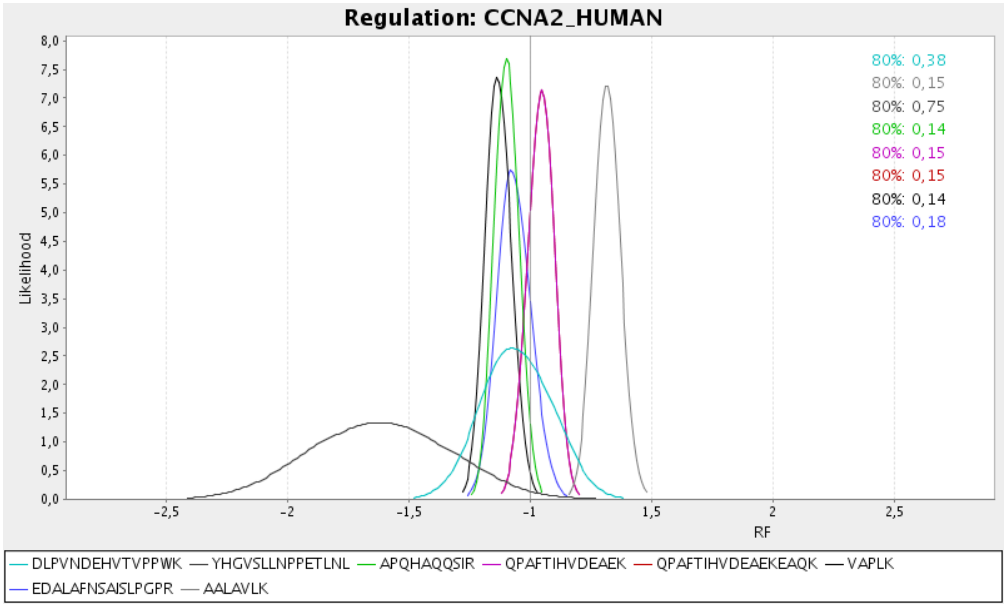


Figure 5.21: Likelihood plot of the protein CCNA2.

Table 5.10: Partition of peptides of the protein CCNA2 into two clusters which were calculated by the presented expectation-maximisation clustering algorithm.

Cluster	Peptides (RF)
1	YHGVSLLNPPETLNL (−1.61)
2	VAPLK (−1.14), APQHAQQSIR (−1.10), EDALAFNSAISLPGPR (−1.08), DLPVNDEHVTVPPWK (−1.07), QPAFTIHVDEAEK(EAQK) (1.05)
3	AALAVLK (1.34)

VII. The protein K0528 (Figure 5.22, Table 5.11) was identified by seven unmodified peptides and one known phosphpeptide (green, RF = 1.05). According to EM clustering the peptides `SQSESSDEVTELDLSHGK` (light purple, RF = −6.21) and `IHNPDEPETR` (dark purple, RF = −2.41) each form separate clusters. Both are not known to be modified. Due to the location of both peptides (positions 379-388 and 657-674 from 1000 amino acids in total) proteolysis can be excluded. Analyses of the both regarding isoforms and wrong assignment are necessary in order to decide whether novel modifications can be assumed and further investigations are started.

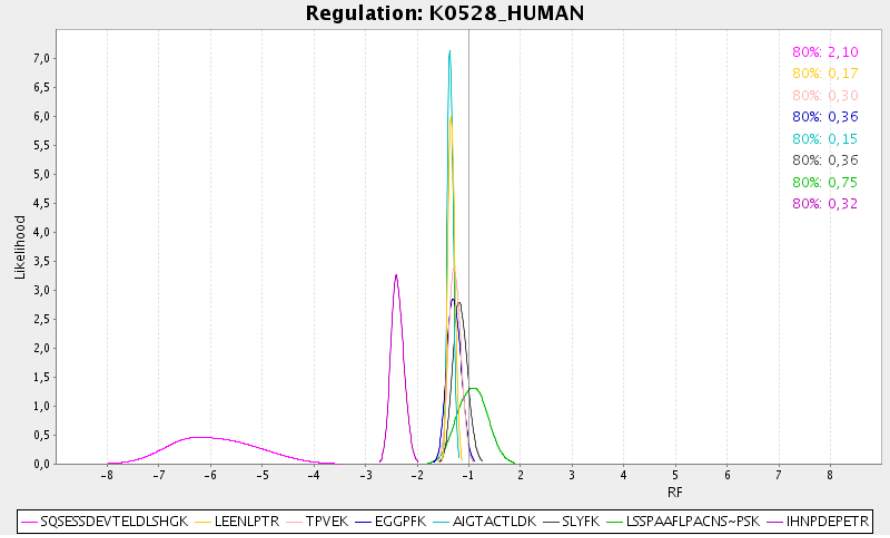


Figure 5.22: Likelihood plot of the protein K0528.

Table 5.11: Partition of peptides of the protein K0528 into four clusters which were calculated by the presented expectation-maximisation clustering algorithm.

Cluster	Peptides (RF)
1	SQSESSDEVTELDLSHGK (−6.21)
2	IHNPDEPETR (−2.41)
3	AIGTACTLDK (−1.37), LEENLPTR (−1.35), EGGPFK (−1.31), TPVEK (1.29), SLYFK (−1.19)
4	LSSPAAFLPACNS∼PSK (1.05)

6 Conclusions and Outlook

The completely novel concept of presenting regulatory information by likelihood curves provides an intuitive and comfortable approach to illustrate both the resulting regulation factor and the underlying data quality. In comparison with long tables containing hundreds and thousands of peptides, their ion intensities and regulation factors, a feasible system was created for analyses of quantitative experiments. The established software tool iTRAQassist combines all components necessary for user-defined analyses comprising data preprocessing, statistically confirmed calculation of regulatory information as well as visualisation of peptide expression profiles, emphasising differently regulated peptides.

The exact calculation of the most probable regulation factor and even more the evaluation of alternative regulation factors is only possible due to the definition of an MS instrument type specific noise model. Although this model was validated carefully, alternative noise models corresponding to alternative workflows might further improve this strategy. If necessary, this workflow allows to substitute the model selected in this work without interfering the following calculations. Essentially, the model used should allow computation of the standard deviation depending on the measured intensity. Adaptability to different workflows also includes customisation to different kinds of (high-throughput) data.

The availability of multiple peptide likelihood curves exhibiting different shapes and different regulations within one protein plot provides a basis for clustering algorithms that are able to detect differentially regulated peptides within one protein in an automatic manner. Investigations of novel post-translational modifications (PTM) are highly topical in biological research and can be supported by clustering analyses in order to detect differentially regulated peptides. A novel clustering algorithm was developed in this work considering properties and characteristics of likelihood curves. Application of this algorithm to experimental data containing regulated phosphopeptides achieved good results regarding the identification of regulated phosphopeptides. Furthermore, several unmodified regulated peptides could be found as well, to identify the causes for the regulations is an open task.

Future work In the meantime, highly productive devices as well as 8 plex iTRAQ^{TM} labelling reagents are available. This provides alternative strategies for the estimation of the model parameters. It should be investigated, whether labelling and combining eight identical treated subsamples could generate enough information to estimate the parameters from a complex sample. Due to powerful devices the number of identified peptides covering a broad intensity range could be increased.

Possibly, the results of the clustering algorithm could be further improved by exclusion of peptides exhibiting extremely low iTRAQ^{TM} reporter intensities. By avoiding the assignment of a single peptide cluster and a very broad curve (resulting from low iTRAQ^{TM} reporter intensities) identification of single peptide clusters could be improved.

Detecting unknown post-translational modifications of peptides not only includes the identification of differentially regulated peptides but requires in-depth investigations of the obtained clustering results. Besides post-translational modifications, several other reasons can cause regulations of peptides, e.g. wrong peptide to protein assignments or the existence of isoforms. By database queries and comparisons with literature as much as possible information must be gathered in order to exclude regulations that are not caused by post-translational modifications or to find further references to PTM interlinking new experimental data with established information. After excluding regulations that can be explained by other reasons than post-translational modifications, additional analyses have to be performed in order to identify the modification type out of more than 200 possible kinds of peptide modifications.

All such regulations happen in time and space. It is therefore of further interest to analyse phosphorylation events in regards to the time of modification and the cell component they took place in. The aim is to explore in-depth the order and rate of the signalling networks that pass on information by activation and deactivation of proteins through PTM. Because of the immense amount of data such a research requires, new automated ways to construct pathways from existing information have to be developed. Again, this can rely on information provided by the proposed clustering algorithm to predict the likeliness of interconnections which can then be validated.

Efforts have been made to continue the results of this work. A new thesis is in preparation which will build onto the clustering results to predict aforementioned network interactions from high-throughput experiments.

List of Abbreviations

1LC	1-Letter-Code
3LC	3-Letter-Code
bp	basepairs
Da	Dalton
DNA	Deoxyribonucleic Acid
e.g.	exempli gratia (for example)
EM	Expectation-Maximisation
ESI	Electrospray Ionisation
HGF	Hepatocyte Growth Factor
HTML	Hypertext Markup Language
i.e.	id est
IR	Interval of Robustness
iTRAQ	Isobaric Tag for Relative and Absolute Quantitation
K-S-Test	Klomogorov-Smirnov-Test
LC	Liquid Chromatography
MLE	Maximum Likelihood Estimation
MS	Mass Spectrometry
PMF	Peptide Mass Fingerprinting
PTM	Posttranslational Modification
Q-Q Plot	Quantile-Quantile Plot
QTOF	Quadrupole Time of Flight
RF	Regulation Factor
RNA	Ribonucleic Acid

List of Figures

List of Tables

References

A.L. Symeonidis, P. M. (2005). *Agent Intelligence Through Data Mining*. Springer, New York.

Anderle, M., Roy, S., Lin, H., Becker, C., and Joho, K. (2004). Quantifying reproducibility for differential proteomics: noise analysis for protein liquid chromatography-mass spectrometry of human serum. *Bioinformatics*, **20**(18), 3575–3582.

Bäck, T. (1996). *Evolutionary Algorithms in Theory and Practise*. Oxford University Press, New York , Oxford.

Baldi, P. and Long, A. D. (2001). A bayesian framework for the analysis of microarray expression data: regularized t -test and statistical inferences of gene changes. *Bioinformatics*, **17**(6), 509–519.

Bezdek, J. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York.

Boehm, A. M., Pütz, S., Altenhöfer, D., Sickmann, A., and Falk, M. (2007). Precise protein quantification based on peptide quantification using itraq. *BMC Bioinformatics*, **8**, 214.

Bolstad, B. M., Irizarry, R. A., Astrand, M., and Speed, T. P. (2003). A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, **19**(2), 185–193.

D'Ascenzo, M., Choe, L., and Lee, K. H. (2008). itraqpak: an r based analysis and visualization package for 8-plex isobaric protein expression data. *Brief Funct Genomic Proteomic*.

Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, **39**(1), 1–38.

Du, P., Stolovitzky, G., Horvatovich, P., Bischoff, R., Lim, J., and Suits, F. (2008). A noise model for mass spectrometry based proteomics. *Bioinformatics*, **24**(8), 1070–1077.

Guojun, G., Ma, C., and Wu, J. (2007). *Data Clustering: Theory, Algorithms, and Applications*. SIAM, Philadelphia.

Hill, T. and Lewicki, P. (2006). *Statistics: Methods and Applications*. StatSoft, Tulsa.

Hu, J., Qian, J., Borisov, O., Pan, S., Li, Y., Liu, T., Deng, L., Wannemacher, K., Kurnellas, M., Patterson, C., Elkabes, S., and Li, H. (2006). Optimized proteomic analysis of a mouse model of cerebellar dysfunction using amine-specific isobaric tags. *Proteomics*, **6**(15), 4321–4334.

Hundertmark, C., Fischer, R., Reinl, T., May, S., Klawonn, F., and Jänsch, L. (2008). Ms-specific noise model reveals the potential of itraq in quantitative proteomics. *Bioinformatics*.

J. Han, M. K. (2001). *Data Mining: Concepts And Techniques*. Morgan Kaufmann, San Francisco.

L. Sachs, J. H. (2006). *Angewandte Statistik*. Springer, Berlin.

Lin, W.-T., Hung, W.-N., Yian, Y.-H., Wu, K.-P., Han, C.-L., Chen, Y.-R., Chen, Y.-J., Sung, T.-Y., and Hsu, W.-L. (2006). Multi-q: a fully automated tool for multiplexed protein quantitation. *J Proteome Res*, **5**(9), 2328–2338.

Lodish, H., Baltimore, D., Berk, A., Zipursky, S.-L., Matsudaira, P., and Darnell, J. (1996). *Molekulare Zellbiologie*. Walter de Gruyter, Berlin.

May, S. (2007). *Webbasiertes System zur Schätzung der Proteinregulation.* Diplomarbeit Fachhochschule Braunschweig/Wolfenbüttel, Fachbereich Informatik.

Perkins, D. N., Pappin, D. J., Creasy, D. M., and Cottrell, J. S. (1999). Probability-based protein identification by searching sequence databases using mass spectrometry data. *Electrophoresis*, **20**(18), 3551–3567.

Pierce, A., Unwin, R. D., Evans, C. A., Griffiths, S., Carney, L., Zhang, L., Jaworska, E., Lee, C.-F., Blinco, D., Okoniewski, M. J., Miller, C. J., Bitton, D. A., Spooncer, E., and Whetton, A. D. (2007). Eight-channel itraq enables comparison of the activity of 6 leukaemogenic tyrosine kinases. *Mol Cell Proteomics*.

Pierce, A., Unwin, R. D., Evans, C. A., Griffiths, S., Carney, L., Zhang, L., Jaworska, E., Lee, C.-F., Blinco, D., Okoniewski, M. J., Miller, C. J., Bitton, D. A., Spooncer, E., and Whetton, A. D. (2008). Eight-channel itraq enables comparison of the activity of six leukemogenic tyrosine kinases. *Mol Cell Proteomics*, **7**(5), 853–863.

Rocke, D. M. and Durbin, B. (2001). A model for measurement error for gene expression arrays. *J Comput Biol*, **8**(6), 557–569.

Ross, P. L., Huang, Y. N., Marchese, J. N., Williamson, B., Parker, K., Hattan, S., Khainovski, N., Pillai, S., Dey, S., Daniels, S., Purkayastha, S., Juhasz, P., Martin, S., Bartlet-Jones, M., He, F., Jacobson, A., and Pappin, D. J. (2004). Multiplexed protein quantitation in saccharomyces cerevisiae using amine-reactive isobaric tagging reagents. *Mol Cell Proteomics*, **3**(12), 1154–1169.

Shapiro, S. S. and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, **3**(52), 591–611.

Tu, Y., Stolovitzky, G., and Klein, U. (2002). Quantitative noise analysis for gene expression microarray experiments. *Proc Natl Acad Sci U S A*, **99**(22), 14031–14036.

W. J. Ewens, G. R. G. (2002). *Statistical Methods in Bioinformatics*. Springer, New York.

Walsh, C. T., Garneau-Tsodikova, S., and Gatto, G. J. (2005). Protein posttranslational modifications: the chemistry of proteome diversifications. *Angew Chem Int Ed Engl*, **44**(45), 7342–7372.

Weng, L., Dai, H., Zhan, Y., He, Y., Stepaniants, S. B., and Bassett, D. E. (2006). Rosetta error model for gene expression analysis. *Bioinformatics*, **22**(9), 1111–1121.

Danksagung

Besonders danken möchte ich Herrn Dr. Lothar Jänsch sowie Herrn Prof. Dr. Frank Klawonn für die hervorragende Betreuung dieser Arbeit sowie ihre ständige Diskussionsbereitschaft.

Des Weiteren bedanke ich mich herzlich bei Herrn Prof. Dr. Jürgen Wehland für die Möglichkeit der Durchführung meiner Doktorarbeit in der Abteilung für Zellbiologie.

Herrn Prof. Dr. Ehrich danke ich für die Übernahme des Korreferates und Herrn Prof. Dr. Wätjen für die Übernahme des Prüfungs-Vorsitzes.

Großer Dank gilt allen Mitgliedern der Arbeitsgruppe CPRO für das sehr gute Arbeitsklima und die ständige Hilfsbereitschaft. Besonders bedanke ich mich bei Dr. Roman Fischer und Kirsten Minkhart für die Unterstützung bei der Validierung des Modelles sowie bei Dr. Manfred Nimtz für hilfreiche Einblicke in die Massenspektrometrie. Thorsten Johl, Dr. Tobias Reinl und Dr. Claudia Pommerenke danke ich für konstruktive Diskussionen.

Bei meiner Familie und allen Freunden bedanke ich mich für ihre Unterstützung und ihr Verständnis während meiner Promotionszeit.